

Návrh a vytvoření demonstrační aplikace pro řízení pohonů pomocí užití programovatelného automatu, vizualizačního systému a průmyslových komunikačních protokolů

Design and creation of a demonstration application for drive control using a programmable controller, visualization system and industrial communication protocols

Tomáš Škuta

Diplomová práce

Vedoucí práce: Ing. Martin Mikolajek

Ostrava, 2021

Abstrakt

Tato práce se zabývá vytvořením řídicí a školící aplikace pro ABB pohon ACS880. Práce bude využívat k řízení B&R X20 Systém, z důvodu jednoduchého propojení za využití řídicího adaptéru FEPL-02, díky kterému je řízení přes B&R systém umožněno. Díky této aplikaci bude možné školit řízení pohonů včetně jejich programování pomocí X20 systému a všeobecné otestování řady těchto pohonů. Bude popsána také technologie DTC, což je technologie od firmy ABB používaná pro řízení pohonů.

Klíčová slova

PLC; ABB; B&R; Pohon

Abstract

This diploma deals with the creation of a control and training application for the ABB ACS880 drive. Diploma will use the B&R System to control the drive, due to the simple connection using the FEPL-02 connection adapter, thanks to which control via the B&R system is enabled. Thanks to this application, it will also be possible to train the control of drives, including their programming using the X20 system and general testing of these drives series. DTC technology, an ABB technology used to control drives, will also be described.

Keywords

PLC; ABB; B&R; Drive

Poděkování

Rád bych poděkoval za odbornou pomoc a konzultaci při vytváření této diplomové práce Ing. Martinu Mikolajkovi a Ing. Tomáši Pitrunovi.

Obsah

Seznam použitých symbolů a zkratk	6
Seznam obrázků	7
Seznam tabulek	9
1 Úvod	10
2 Popis ABB pohonu ACS880, X20 B&R Systému, adaptéru FEPL-02 a Ethernet POWERLINK	12
2.1 Pohon ACS880	13
2.2 X20 B&R Systém	19
2.3 Adaptér FEPL-02	26
2.4 Ethernet POWERLINK	27
3 Popis komunikace ACS880 a B&R PLC pomocí FEPL-02 a Ethernet POWERLINK	29
4 Návrh cílů pro testovací řídicí aplikaci za pomoci využití systému ACS880 a automatu B&R	33
5 Popis konfigurace systému a realizace programu pro navrženou aplikaci	35
5.1 Konfigurace	36
5.2 Topologie	38
5.3 Propojení ACS880 s B&R automatem	40
5.4 Tvorba a popis řídicí aplikace	45
5.5 Popis řídicí aplikace - Lokální řízení	49
5.6 Popis řídicí aplikace - Dálkové řízení	50
6 Návrh a vytvoření vizualizace pro řízení pohonu	57
6.1 Konfigurace	57

6.2	Sledované a řízené proměnné díky aplikaci a její vizualizaci	58
6.3	Vizualizace - Lokální řízení	59
6.4	Vizualizace - Dálkové řízení	60
7	Zhodnocení dosažených výsledků a možnosti uplatnění realizované aplikace	63
	Literatura	65
	Přílohy	66
A	Řídící aplikace (Automation Studio)	67

Seznam použitých zkratk a symbolů

PLC	– Programovatelný logický kontrolér
DTC	– Direct Torque Control - Přímé Řízení Momentu
PWM	– Pulzně šířková modulace
TCP/IP	– Transmission Control Protocol/Internet Protocol - Primární přenosový protokol/protokol síťové vrstvy
HMI	– Human Machine Interface - Rozhraní mezi člověkem a strojem
LAN	– Local Area Network - Lokální síť
CTU	– Count Up - Čítač nahoru
CTUD	– Count Up/Down - Čítač nahoru/dolů
RS	– Reset Set - Klopný obvod pro resetování či nastavení stavu
TOF	– Timer off Delay - Časovač se zpožděním při vypnutí
TON	– Timer on Delay - Časovač se zpožděním při zapnutí
CU	– Count Up - Přičítání nahoru
ASC	– Auto Speed Control - Automatické řízení rychlosti
LED	– Light-Emitting Diode - Světelná dioda
RPM	– Revolutions Per Minute - Otáčky za minutu
I/O	– Input/Output - Vstup/Výstup
PID	– Proporcionální, integrační, derivační
UDP	– User Datagram Protocol - Protokol uživatele diagramu
IEEE	– Institute of Electrical and Electronics Engineers - Institut elektrických a elektronických inženýrů
IEC	– International Electrotechnical Commission - Mezinárodní elektrotechnická komise

Seznam obrázků

2.1	Demokufr s pohonem ACS880, motorem a I/O panelem včetně řídicího panelu [2] . .	13
2.2	Panel pro nastavení parametrů na pohonu [1]	14
2.3	Porovnání DTC technologie s PWM [6]	15
2.4	Princip DTC technologie [5]	18
2.5	Azipod [7]	19
2.6	X20 Systém - I/O Karty a jejich modulace [8]	20
2.7	I/O karta a její části [8]	21
2.8	Procesory [8]	22
2.9	LED diody pro diagnostiku [8]	23
2.10	System Diagnostics Manager [11]	25
2.11	Adaptér FEPL-02 [12]	26
2.12	Ethernet POWERLINK [13]	28
3.1	Kontrolní slovo [12]	30
3.2	Stavové slovo [12]	31
3.3	Schéma komunikačního profilu pro řízení pohonu ACS880 [12]	32
4.1	Návrh školícího pracoviště	34
5.1	Konfigurace aplikace	37
5.2	Topologie lokálního řízení	38
5.3	Topologie dálkového řízení	39
5.4	Schéma propojení automatu s pohonem [12]	40
5.5	Nastavení parametrů na řídicím panelu, umístěném na pohonu	41
5.6	Nastavení parametrů v aplikaci Drive Composer	42
5.7	Nahrání XDD souboru do Automation Studio	44
5.8	Přípojení pohonu na Ethernet POWERLINK	44
5.9	Monitorování hodnot na pohonu	45
5.10	Napojení vytvořené proměnné na hodnotu z pohonu - Vstup	46

5.11	Napojení vytvořené proměnné na hodnotu z pohonu - Výstup	46
5.12	Konfigurace pro lokální řízení	49
5.13	Konfigurace pro dálkové řízení	51
5.14	Část programu se základní spouštěcí sekvencí	54
5.15	Příklad alarmu a jeho nastavení v aplikaci	56
6.1	Vizualizace pro lokální řízení	59
6.2	Konfigurace vizualizace pro dálkové řízení	60
6.3	Vizualizace pro dálkové řízení - řídicí displej	61
6.4	Vizualizace pro dálkové řízení - Vizualizační displej	62

Seznam tabulek

5.1	Dostupné komponenty obsažené v konfiguraci aplikace	36
5.2	Parametry pro řízení pohonu - Skupina 50 a její podskupiny[12]	42
5.3	Parametry pro řízení pohonu - Skupina 51 a její podskupiny[12]	43
5.4	Parametry pro řízení pohonu - Skupina 20 a její podskupiny[12]	43
5.5	Parametry pro řízení pohonu - Skupina 22 a její podskupiny[12]	43
5.6	Proměnná pro kontrolní a stavové slovo s popisem jejich funkce v aplikaci	46
5.7	Proměnné s popisem jejich funkce v aplikaci	47
5.8	Funkční bloky s popisem jejich funkce v aplikaci	48
6.1	Funkční bloky s popisem jejich funkce v aplikaci	58

Kapitola 1

Úvod

Tato práce se zabývá vytvořením aplikace a komunikace pro řízení a monitorování ABB pohonu ACS880 pomocí X20 B&R systému, adaptéru FEPL-02 a Ethernet POWERLINK, včetně jejich propojení. Práce se aplikací zabývá z důvodu potřeby školícího stanoviště ve firmě, aby každý nový pracovník měl možnost si vyzkoušet jak se tyto pohony řídí, programují a získají celkový nadhled nad problematikou. Aktuálně se v místě využití této aplikace nachází podobná aplikace, avšak využívající jiné komunikační standardy, automat od firmy ABB a pohon starší řady, který se řídí jiným způsobem.

Vytvořená aplikace je tedy určena pro demonstraci funkce pohonu a samotného řízení či monitorování pomocí PLC. Slouží k účelu školení, programování a ovládání vybraného pohonu pomocí příslušného komunikačního profilu. Tento komunikační profil je možno si zvolit díky adaptéru FEPL-02. Mimo programování a řízení pohonu bude díky ní možno také školení samotného programování PLC včetně řídicích a vizualizačních panelů a aplikací.

Pro řízení je využit procesor z již zmíněného X20 Systému, a to X20CP3586. Tento automat komunikuje s pohonem pomocí adaptéru FEPL-02. Pro demonstraci funkce pohonu aplikace využívá ACS880 Demo Case, který je kromě samotného pohonu vybaven také I/O panelem – který je naprogramován s využitím pro lokální řízení (řízení pomocí PLC bude využito jako dálkové řízení) a samozřejmě střídavým motorem také od výrobce ABB, který je vybaven mechanickou brzdou pro simulaci zátěže. Všechny využití komponenty jsou popsány v teoretické části, včetně zajímavosti v případě pohonu a to popis technologie Azipod.

Řídící aplikace je naprogramována do dvou režimů. První je klasické řízení rychlosti, kde uživatel zadá požadovanou rychlost a spustí/zastaví/vy-resetuje pohon. Druhý režim je automatické periodické navyšování rychlosti na její maximum a poté zpátky na nulu pro zobrazení průběhů sledovaných hodnot v širším spektru. Aplikace je řízena pomocí vizualizačních panelů, kdy jeden je použit jako řídicí displej pro ovládací prvky a druhý čistě jako vizualizační. Aplikace včetně vizualizací je popsána v praktické části této práce.

Cílem kromě řídicí aplikace, jejího popisu, demonstrace ovládání a programování X20 Systému je také ukázka Direct Torque Control (DTC) technologie, kterou pohon využívá a její porovnání s PWM včetně vlivu zátěže (brzdění) na moment motoru. Kromě toho je pohon používán také na lodích ve větším provedení, tudíž dalším cílem je popsání parametrů, které je potřeba řídit, monitorovat a jakým způsobem se na lodích řídí.

Kapitola 2

Popis ABB pohonu ACS880, X20 B&R Systému, adaptéru FEPL-02 a Ethernet POWERLINK

V této kapitole je popsán pohon ACS880, který je programován, X20 B&R systém, na kterém běží řídicí aplikace, adaptér FEPL-02, díky kterému je propojení umožněno a protokol Ethernet POWERLINK. V práci je využíván demokufr, ve kterém se nachází jak samotný pohon, motor a také I/O panel.

2.1 Pohon ACS880

Jedná se o pohon od firmy ABB. Jedná se o velice kompatibilní průmyslový pohon. Vyrábí se ve variantách jako vestavěné pohony, modulové pohony a také v kabinových variantách, používaných především na lodích. Tyto pohony se používají v nejrůznějších odvětvích průmyslu. Například zpracování materiálů, papírnictví, lodní průmysl a mnoho dalších odvětví. Dokážou řídit širokou škálu aplikací včetně jeřábů, pásových dopravníků a Azipodů. Pohon je umístěn v demo kufru společně s motorem, I/O panelem a včetně brzdy pro simulaci zátěže jak lze vidět na obr. 2.1. [1]



Obrázek 2.1: Demokufř s pohonem ACS880, motorem a I/O panelem včetně řídicího panelu [2]

Tyto pohony mají vysokou kvalitu zpracování pro funkčnost v nejnáročnějších podmínkách. Nejen pro funkčnost, ale také pro co největší spolehlivost a efektivnost. Všechny tyto pohony - včetně ACS880, jsou vybaveny jedinečnou technologií od ABB, která se nazývá DTC - Direct Torque Control. Díky této technologii je dosaženo velice přesného řízení rychlosti a momentu. Technologie je popsána v další podkapitole, včetně jejího principu a blokového schématu. Pohony jsou také vybaveny vším potřebným pro rychlou instalaci, údržbu a širokou škálu různých požadavků, včetně certifikovaných bezpečnostních požadavků. Všechny pohony této řady sdílí stejné uživatelské rozhraní v podobě displeje, který se jednoduše připojí přímo na pohon a probíhá na něm nastavení veškerých parametrů - obr. 2.2. Tyto parametry je tak možno nastavit v programu Drive Composer, který je stvořen pro virtuální commissioning - což je testování a rozběhnutí zařízení. Virtuální commissioning je poté využíván v části práce, kde dochází k propojení automatu s pohonem a jeho nastavení - přesněji jeho parametrů pro povolené a nastavení komunikace mezi pohonem a automatem. [1]



Obrázek 2.2: Panel pro nastavení parametrů na pohonu [1]

Další velice užitečnou vlastností tohoto pohonu a jeho připojitelného panelu k nastavení je ta, že se k němu dá připojit pomocí mobilního telefonu, kde lze provést tyto úkony:

- Nastavení parametrů
- Záloha a obnovení funkcionality
- Přístup k datům z pohonu a historie servisu
- Sdílení konfigurace přes email či Bluetooth

2.1.1 Drive Composer

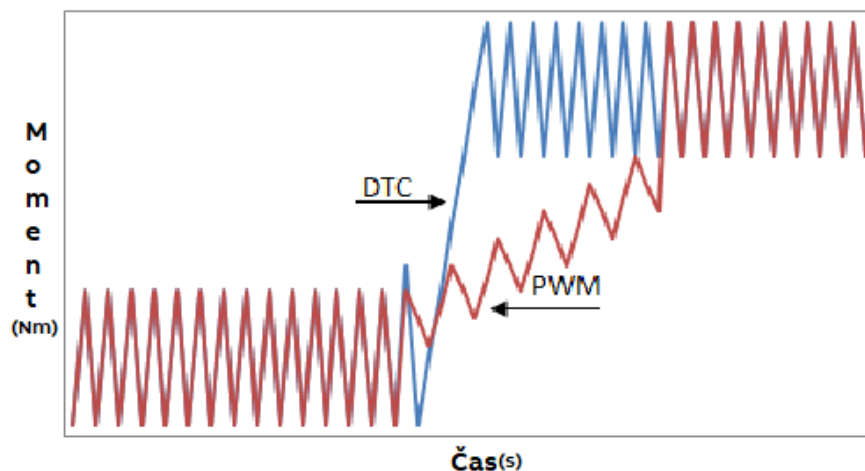
Je nástroj pro spuštění a údržbu běžných pohonů architektury ABB. Nástroj se používá k prohlížení a nastavování parametrů k pohonu, včetně jejich monitorování a ladění. Dále se v něm dá naprogramovat I/O panel a použít blokové programování k sestavení logiky řízení pohonu bez PLC. [3]

2.1.2 Řízený motor - M2VA63B4 0,18KW 4P V.380Y/50HZ B14

Řízený motor pohonem v demo kufru je nízkonapěťový střídavý motor od firmy ABB. Jedná se o motor s prodloužením hřídele vybaveným brzdou pro simulaci zátěže na motoru. Jeho výkon je 0.18kW. [4]

2.1.3 DTC - Direct Torque Control

Jedná se o přímé řízení točivého momentu, používané v pohonech k řízení právě točivého momentu a rychlosti v třífázových střídavých elektromotorech. Tato technologie zahrnuje výpočet odhadu magnetického toku a momentu na základě naměřeného napětí a proudu motoru. DTC dosahuje požadovaného momentu daleko rychleji než PWM - Pulzně šířková modulace. Tato skutečnost je zobrazena na obr. 2.3. V tomto grafu je vyobrazena závislost momentu na čase, který dosáhne požadované hodnoty mnohem rychleji při použití DTC.[5]



Obrázek 2.3: Porovnání DTC technologie s PWM [6]

Většina výkonných pohonů se v 80. letech spoléhala na technologii PWM. Při této technologii je třeba filtrovat měřené proudy při provádění řízení motoru - tedy zpomalení odezvy točivého momentu motoru. Právě proto ABB zaujalo odlišný přístup k vysoce výkonnému řízení střídavých motorů. Střídavé pohony od ABB určené pro náročné aplikace používají právě tento inovativní způsob přímého řízení točivého momentu. Jak již bylo zmíněno, metoda přímo ovládá točivý moment motoru místo toho, aby se snažil ovládat proudy analogicky k stejnosměrným pohonům. To znamená větší přesnost ve shodě s požadavky na zatížení poháněného systému. Technologie dále eliminuje potřebu další fáze modulátoru a tím dosahuje dynamika řízení teoretického maxima. [5]

Tato technologie vznikla v polovině 80. let, kdy byla i patentována. První pohon s touto technologií byl pak představen na trhu v roce 1995.

Výhody používání Direct Torque Control:

- Není potřeba změna otáček motoru ani polohové zpětné vazby v 95% aplikací. Tedy instalaci nákladných kodérů nebo jiným zpětnovazebním zařízením se lze vyhnout.
- Řízení DTC je k dispozici pro různé typy motorů, včetně permanentního magnetu a synchronního motoru.
- Přesné řízení točivého momentu.
- Vynikající linearita točivého momentu.
- Vysoká statická a dynamická přesnost rychlosti.
- Žádná přednastavená spínací frekvence. [5]

Z širšího pohledu se výhody DTC rozšiřují na software, uživatelské rozhraní, údržbu a funkce systému. Jak již bylo řečeno a název napovídá, DTC se snaží řídit tok motoru a točivý moment přímo, místo toho, aby se snažil tyto proměnné ovládat nepřímo jako stejnosměrné pohony a vektorově řízené střídavé pohony. Samostatné smyčky řízení točivého momentu a rychlosti tvoří maximum. Systém DTC spolupracuje integrovaným způsobem.

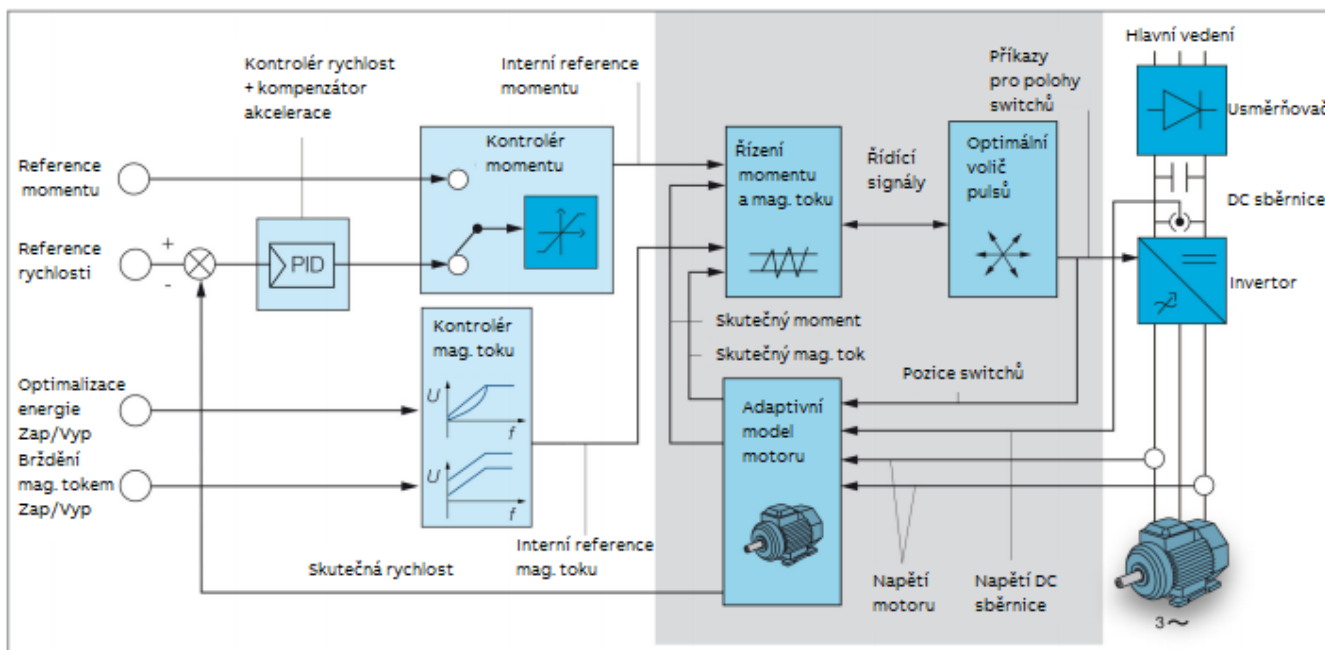
Princip funkce DTC je poměrně jednoduchý. Na obr. 2.4. je vidět, že jádrem je smyčka řízení točivého momentu, kde je sofistikovaný adaptivní model motoru, který využívá pokročilé matematické algoritmy pro predikci stavu motoru. Primárně řízené proměnné jsou magnetický tok statoru a točivý moment motoru. Model tyto proměnné odhaduje pomocí vstupů fáze motoru, kde měří proudy, napětí střídavé sběrnice a stavy výkonových spínacích tranzistorů v měniči. Model motoru také počítá otáčky hřídele. Přítomná teplotní kompenzace pomáhá zvýšit přesnost výpočtu bez jakéhokoli kodéru.

Další parametry motoru jsou automaticky přiváděny na adaptivní model během identifikace motoru, když je pohon uváděn do provozu. V mnoha případech lze vhodnou identifikací parametrů modelu provést bez otáčení hřídele motoru. Pro jemné doladění motoru musí být však motor spuštěn, ale pouze velmi krátce a bez zátěže.

Odpor statoru (pokles napětí) je jediný a snadný měřitelný parametr potřebný pro odhad magnetického toku. Kroutící moment motoru lze poté vypočítat jako křížový produkt odhadovaného toku statoru a statorových vektorů. Zatímco statorový odpor je hlavním zdrojem chyby odhadu, jeho vliv se zmenšuje se zvyšující se rychlostí a napětí motoru. DTC má tedy vynikající točivý moment a přesnost v širokém rozsahu rychlostí. Navíc DTC zahrnuje pokročilé způsoby minimalizace chyby odhadu při nízkých otáčkách motoru.

Výstupní signály z modelu motoru - které představují skutečný magnetický tok statoru a točivý moment motoru - přejdou na kontrolér magnetického toku a kontrolér točivého momentu. Tyto samostatné řídicí jednotky porovnávají své vstupy s referenční hodnotou magnetického toku a točivého momentu. Tyto funkce jsou prováděny každých 12.5 mikro sekund - což znamená vysoký výkon řízení. Oba tyto kontroléry hledají příslušné velikosti vektoru toku nebo momentu v úzkém hysterézním pásmu kolem referenční hodnoty. Výjimečná odezva motoru je způsobena řídicím algoritmem aktualizující adaptivní model motoru se stejnou rychlostí cyklu - velmi vysokou.

Chyby toku a momentu - rozdíly mezi odhadovanými a referenčními hodnotami se používají k výpočtu stavu toku a momentu v hysterézních regulátorech. Poté se tyto stavové hodnoty stanou vstupy do pulzního voliče, kde je vybrán optimální vektor napětí. Takto lze posílat nejvhodnější signálové pulzy pro každý řídicí cyklus na výkonové spínače ve střídači a získat nebo udržovat přesný točivý moment motoru. [5]



Obrázek 2.4: Princip DTC technologie [5]

2.1.4 Azipod

V lodním průmyslu v případě kdy se o projekt stará ABB lodní pohony pohání Azipody. Jedná se elektricky poháněný azimutový propeler vyvinutý společností ABB. Jedná se o námořní pohonnou jednotku skládající se z vrtule s pevným stoupáním na říditelné gondole, která obsahuje elektromotor pohánějící vrtuli.

V tradičních tryškách s azimutem je vrtule poháněna elektromotorem nebo dieselovým motorem uvnitř trupu lodi. Vrtule je spojena s hnacím motorem pomocí hřídelí a kuželových převodů, které umožňují otáčení vrtule kolem svislé osy. Tento typ pohonného systému má v průběhu 90. let dlouhou tradici a dnes takové pohonné jednotky vyrábí celá řada společností po celém světě.

V jednotce Azipod je však elektromotor namontován uvnitř pohonné jednotky a vrtule je připojena přímo k hřídeli motoru. Elektrická energie pro pohonný motor je vedena přes sběrací kroužky, které umožňují jednotce Azipod otáčet se o 360 stupňů kolem svislé osy. Protože jednotky Azipod využívají vrtule s pevným stoupáním, je energie vždy napájena pohonem s proměnnou frekvencí nebo cyklokonvertorem, který umožňuje řízení rychlosti a směru pohonných motorů.

Vrtule obvykle směřuje dopředu, protože v této tažné konfiguraci je vrtule efektivnější díky provozu v nerušeném proudu. Vzhledem k tomu, že se může otáčet kolem osy připojení, může Azipod aplikovat svůj tah v libovolném směru. Azimutové trysky umožňují lodím lépe manévrovat a umožňují jim cestovat dozadu téměř stejně efektivně, jak mohou cestovat vpřed.

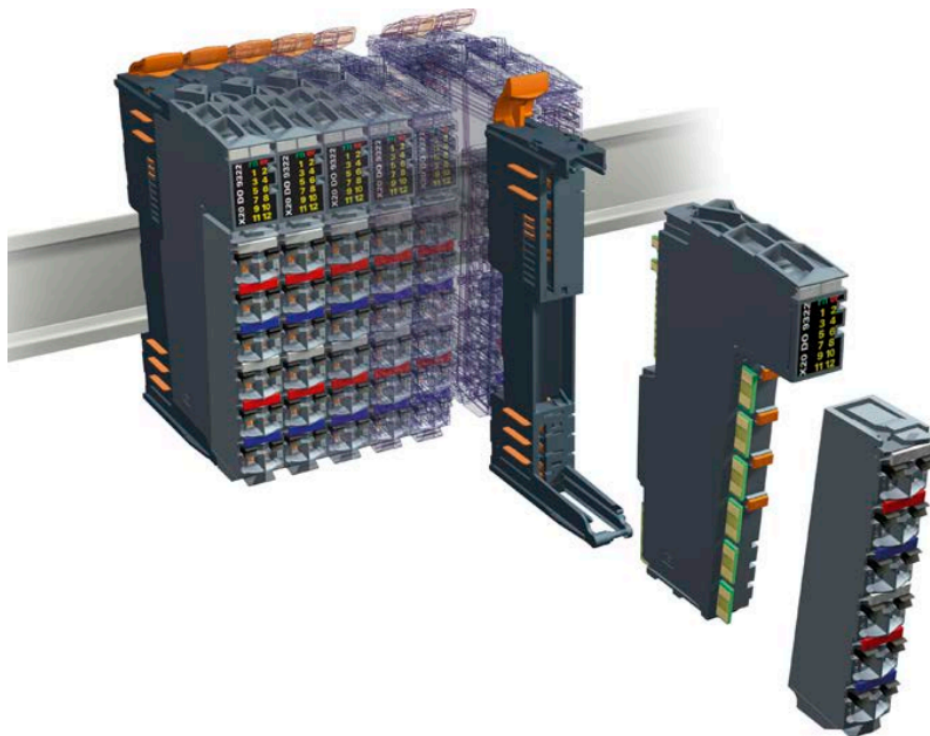
Při první instalaci v 90. letech dosáhla konstrukce obvykle o 9% lepší palivové účinnosti než konvenční pohonný systém. Vylepšení konvenčního designu zmenšily mezeru na 6% - 8%, ale na druhé straně byl hydrodynamický tok kolem Azipodu vylepšen dodatečnými úpravami žeber a dynamickou počítačovou optimalizací příslušných provozních úhlů v instalacích s více zařízeními, s celkovým zvýšením účinnosti nyní v rozmezí 18%. Dnes se díky této technologii ušetří obrovské množství energie, což pomáhá k šetření životního prostředí. [7]



Obrázek 2.5: Azipod [7]

2.2 X20 B&R Systém

Programovatelné logické automaty včetně zdrojů napájení byly navrženy a vyvinuty společností B&R jak pro konvenční užití, tak pro užití v průmyslu se zvýšenými bezpečnostními požadavky. Jedná se o automaty využívané na všech polích průmyslu i mimo průmysl, díky jejich univerzálnosti pro jakoukoliv řídicí úlohu. Na trhu se již nachází velké množství řídicích systémů. Ovšem X20 systém pokračuje v nastavování standardů odkazující se na motto "Perfection in Automation"- v překladu "Perfektní v Automatizaci". B&R se snaží díky zkušenostem z aplikací po celém světě zajistit snadnou, bezpečnou a ekonomicky nenáročnou cestu řízení. V rámci řízení jsou k dispozici také vizualizační panely, některé mají dokonce svůj vlastní procesor. Je díky nim možno samostatně řídit menší úlohy. Klasické vizualizační panely pak slouží ke komunikaci a vizualizaci dat z procesoru. [8]

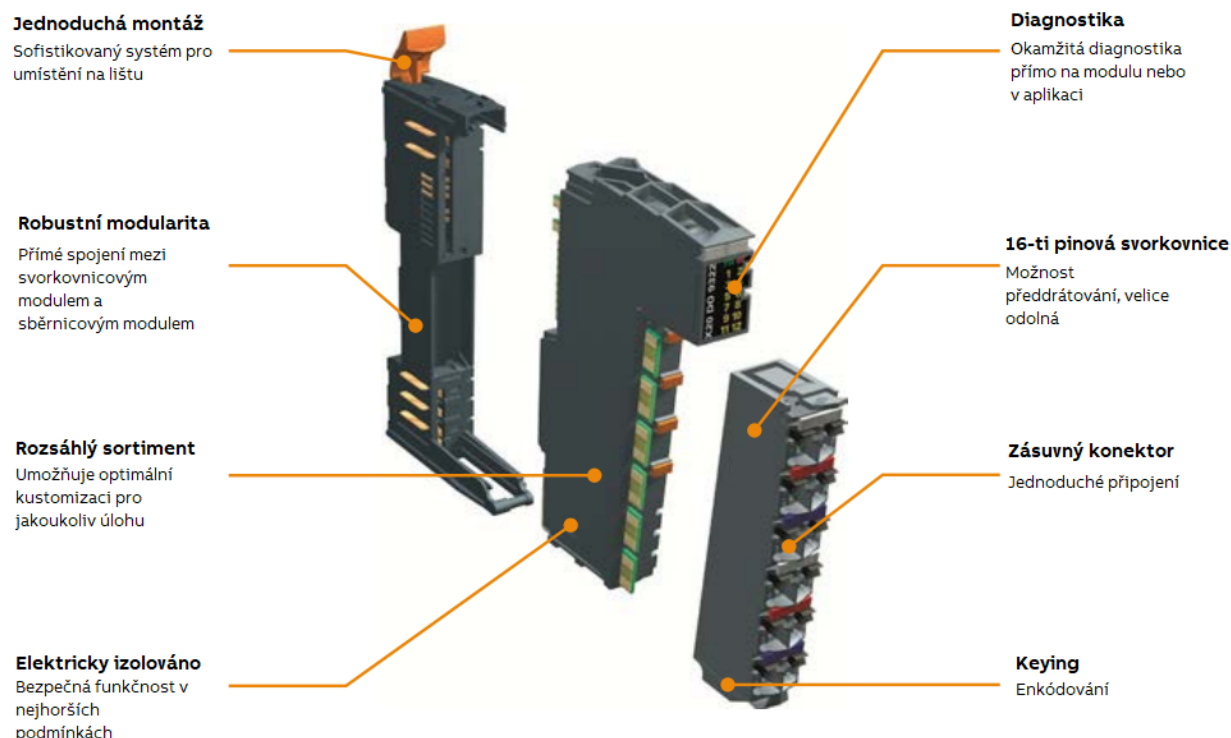


Obrázek 2.6: X20 Systém - I/O Karty a jejich modulace [8]

Díky dobře promyšleným detailům, sofistikovanému a ergonomickému designu, je systém více než řídicí systém - je to kompletní řídicí řešení. X20 systém umožňuje kombinovat přesné komponenty ke splnění jakýchkoliv požadavků aplikace. Jsou ideálním doplňkem standardní sběrnice a rozšiřují možnosti konvenčních řídicích systémů. Stačí je jednoduše připojit, nakonfigurovat a je hotovo. Ve spojení s dalšími komponenty B&R dosahuje X20 systém svého plného potenciálu a umožňuje implementaci aplikací s nepředstavitelným výkonem a flexibilitou. Tento typ bezproblémové integrace je hlavní výhodou. [8]

2.2.1 I/O Moduly, X2X Link

Vstupní či výstupní karty systému se skládají ze tří částí. Ze svorkovnice, elektrického modulu a sběrnice. Díky této modularitě může dojít k před drátováním modulu, je možno připojit elektroniku za provozu a k dispozici jsou extra sběrnice sloty pro další možnosti řešení. Další velkou výhodou je, že Automation Studio, což je software pro vytváření aplikací pro systém včetně jejich diagnostiky, může načíst konfiguraci modulů do vývojového prostředí automaticky. Důležitá je snadná výměna komponentu pro zjištění případně závady modulu nebo jeho části. [8]



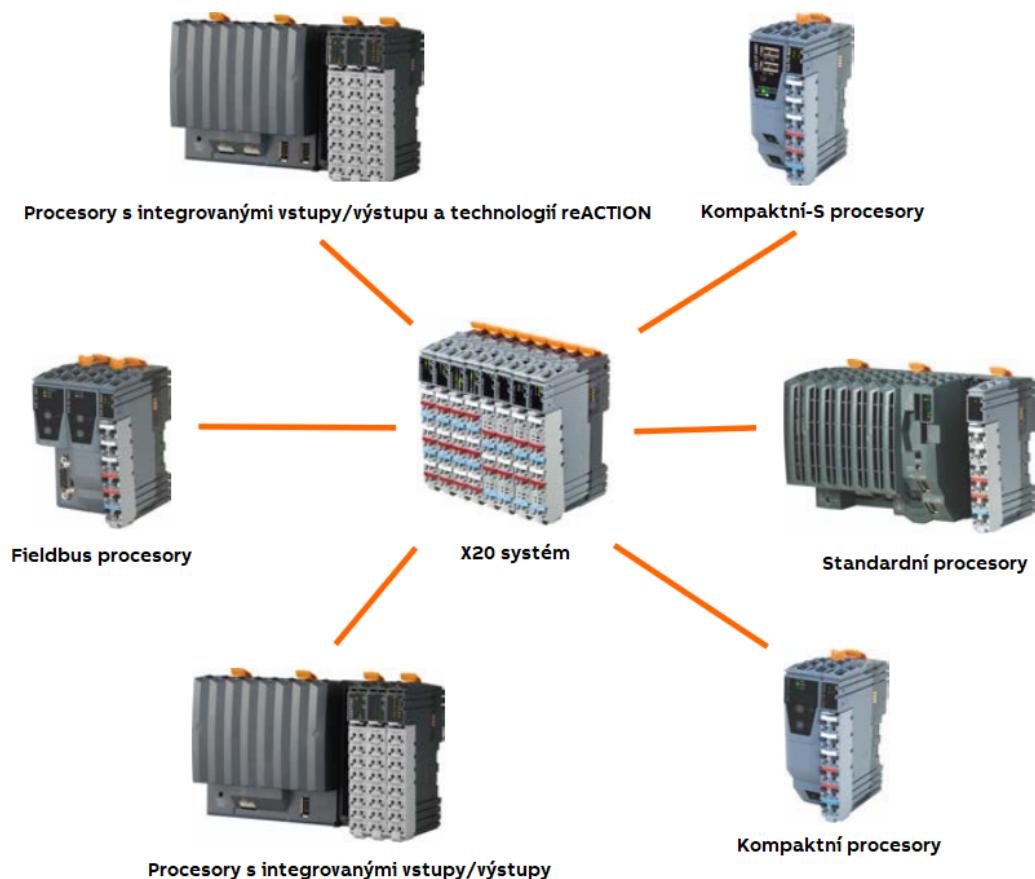
Obrázek 2.7: I/O karta a její části [8]

Další skvělou výhodou je X2X Link. Jedná se o propojení všech modulů pomocí kontaktů na boční straně těchto modulů. Díky tomuto Linku není třeba moduly mezi sebou nijak propojovat. Stačí je pouze umístit na lištu a zacvaknout do sebe. Moduly mimo X2X Link mohou být vzdáleny maximálně 100 metrů pro přímé připojení. X2X Link umožňuje nejvyšší možnou odolnost proti rušení na základě kroucených měděných kabelů.

Tato technologie neposkytuje pouze univerzální propojovací prvek, který se stará o komunikace mezi moduly, ale hlavně zařizuje propojení bez jakýchkoliv konvertorů nebo ztráty výkonu a bez potřeby softwarově měnit adresování.[8]

2.2.2 Procesory

Řada procesorů v X20 systému pokrývá širokou škálu požadavků. Oblasti použití zahrnují jednoduché aplikace, kde jsou doby cyklu v milisekundovém rozsahu dostatečné pro aplikace, které kladou nejvyšší požadavky na výkon. V těchto případech lze efektivně využít i dobu cyklu 100 mikrosekund. Konstrukce procesorů je v souladu s X20 systémem. X20 moduly jsou připojeny přímo k procesoru pomocí již zmíněného X2X linku. Zdroj napájení, X2X link síť a I/O moduly jsou součástí procesoru. Žádné další zdroje napájení nejsou potřeba. [8]



Obrázek 2.8: Procesory [8]

Procesory jsou založeny na ATOM procesor technologii. Všechny jsou také vybaveny USB, Ethernetem, Powerlink V1/V2 a vyjmutelnou CompactFlash kartou pro nahrávání aplikace. Standardní Ethernet rozhraní je schopné řídit gigabitovou komunikací. Procesory mají navíc tři volné sloty pro další moduly ke zvýšení flexibility.

Parametry a výhody procesorů:

- CPU frekvence hodinového signálu od 100 do 1600 MHz
- Ethernet, Powerlink V1/V2, USB
- 3 Sloty pro další moduly
- CompactFlash karta pro aplikaci
- 512 MB DDR2 SRAM
- Pasivní chlazení [8]

2.2.3 Programování - Automation Studio

Programování systému probíhá v aplikaci B&R Automation Studio. Lze použít všechny programovací jazyky relevantní k IEC 61131-3 normě a také jazyk C. Pomocí tohoto ústředního nástroje je programován X20 systém. V aplikaci lze provádět kromě programování také diagnostiku, tvorba vizualizace a programování safety PLC. Jedná se velmi intuitivní a jednoduchý software. Je to jediná aplikace potřebná pro kompletní programování X20 systému. Aplikace obsahuje také velmi užitečný help, který slouží jako datasheet pro všechny moduly v celém systému. [8]

2.2.4 Jazyk C

V této práci byl využit jazyk C. *"V současné době je to jeden z nejpobulárnějších jazyků, zřejmě nejčastější pro psaní systémového softwaru, ale velmi rozšířený i pro aplikace. C je nízkourovňový, kompilovaný, relativně minimalistický programovací jazyk. Je dostatečně mocný na většinu systémového programování (ovladače a jádro OS), přičemž zbytek lze dořešit tzv. inline assemblerem, tedy metodou zápisu assembleru přímo do kódu. Zdrojový kód C je přitom mnohem čitelnější než assembler, je jednodušší ho zapsat a navíc je daleko snáze přenositelný na jiné procesory a počítačové architektury. Proto jsou často operační systémy, překladače, knihovny a interprety vysokoúrovňových jazyků implementovány právě v C. Mnoho dalších moderních programovacích jazyků přebírá způsob zápisu (neboli syntaxi) z jazyka C, například C++, Java, Perl a PHP."* [9]

2.2.5 Diagnostika na modulech

Pro rychlé nalezení chyb systém umožňuje nejlepší diagnostické řešení. X20 systém obsahuje několik úrovní diagnostiky. Prvním z nich jsou LED diody přímo na modulu. Zobrazují stav sběrnice, stav modulu, stav vstupů či výstupů a stavy jednotlivých kanálů a to přímo na modulu jak je vyobrazeno na obr. 2.9 níže. Stavy jsou vyobrazeny pomocí barev, či sekvencí blikání. Všechny tyto barvy a sekvence jsou vysvětleny v dokumentaci každého modulu.



Obrázek 2.9: LED diody pro diagnostiku [8]

Dále dochází k diagnostice za pomoci využití Automation Studia. Data jsou pořád cyklicky přenášena a zobrazována v aplikaci. Poslední možnost diagnostiky je pomoci použití asynchronního kanálu.

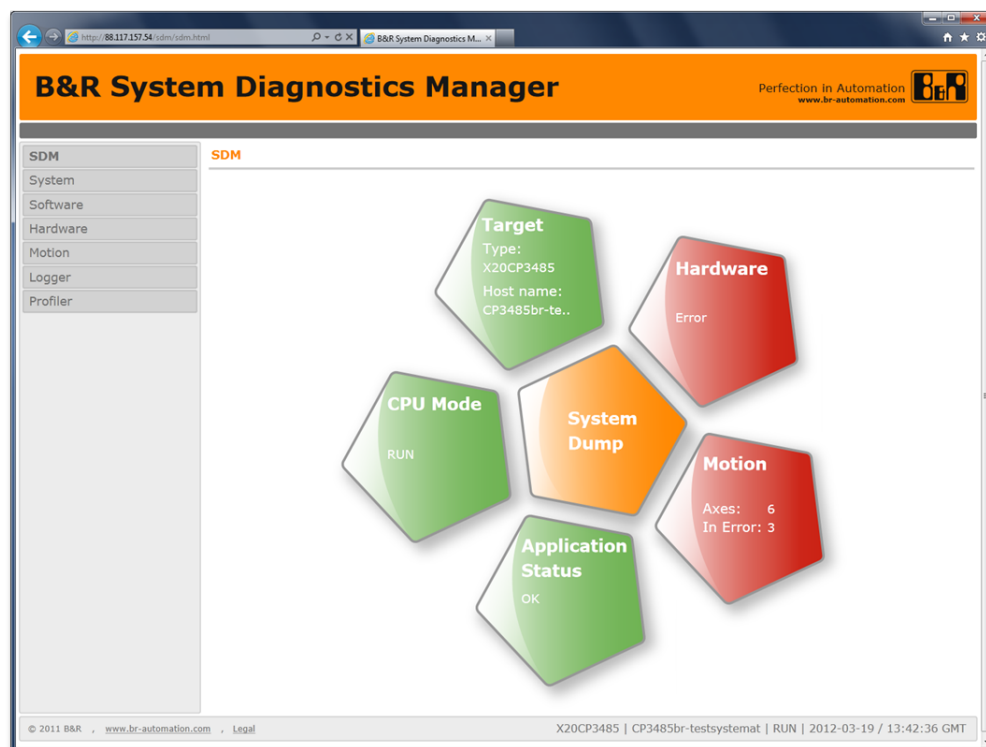
Informace jako typ modulu, sériové číslo, funkcionalita a číslo verze jsou uložena v parametrovém čipu samotného modulu. Tyto informace jsou automaticky evaluovány programovací aplikací (Automation Studio). Tato funkce je prevencí před chybami, které by mohly nastat při commissioningu či při servisování. [8]

2.2.6 System Diagnostics Manager

Další úroveň diagnostiky je systémový manažer diagnostiky, který je obsažen v Automation Studiu a slouží k diagnostice kontroléru ve standardním webovém prohlížeči odkudkoliv - z Intranetu či Internetu. Díky tomuto nástroji již není třeba analyzovat kontrolér speciálními nástroji pro diagnostiku. Stačí pouze již zmíněný webový prohlížeč a TCP/IP připojení ke kontroléru. [10]

Tento nástroj umožňuje:

- Analyzovat Hardware pro detekci konfiguraci či hardwarových chyb v systému
- Analyzovat systémovou konfiguraci a parametry systému (IP adresa apod.) a provádět analýzu Softwaru
- Zobrazit užitečné hodnoty a informace o procesoru - např. teplotu
- Zobrazit a nahrát chybové záznamy [10]



Obrázek 2.10: System Diagnostics Manager [11]

2.3 Adaptér FEPL-02

Modul FEPL-02 Ethernet POWERLINK je volitelné zařízení pro pohony ABB, které umožňuje připojení pohonu do sítě Ethernet POWERLINK. Modul je klasifikován jako plnohodnotný Ethernet POWERLINK slave. Pomocí adaptéru je možno:

- Vydávat řídicí povely pro pohon - Start, Stop apod.
- Poslat referenci rychlosti či momentu
- Uvádět skutečné hodnoty procesu nebo odkaz na proces k PID regulátoru
- Číst informace o stavu včetně skutečných hodnot z měniče
- Změnit hodnoty parametrů pohonu
- Resetovat poruchu pohonu



Obrázek 2.11: Adaptér FEPL-02 [12]

Modul adaptéru je nainstalován do libovolného slotu na jednotce pohonu. Řízení pomocí tohoto pohonu je popsáno v kapitole o komunikaci mezi kontrolérem a pohonem. [12]

2.4 Ethernet POWERLINK

Jedná se o průmyslový standard Ethernetu, který dokáže zařídit přenosy dat v reálném čase. To základní standard nezvládne. Nestací ani žádné doplňkové metody jako například prioritizace rámců. Hlavně se jedná o nezvládnutí požadavku na flexibilitu topologie automatizačních sítí, nemožnost zaručení deterministických přenosů dat a přesnost časování. Nejsložitější je pak konfigurování využívání sítě, nastavováním priorit uzlů a rámců.

Proto mnoho uživatelů a vývojářů vymyslelo nové mechanismy pro zajištění Ethernetu v reálném čase. Patří k nim přístupové mechanismy na fyzické médium využívající ethernetová rámce, které se musí cyklicky dekodovat pomocí speciálních součástí nebo pomocí zkracování přenosových dob.

Protokol POWERLINK je vyvíjen od počátku tak, aby byly dodrženy všechny normy. Jde o otevřený standard a potřebuje rozšíření pro reálný čas, která nebudou závislá na jedné společnosti. POWERLINK rozšiřuje Ethernet podle všech norem – zejména IEEE 802.3 o mechanismus pro rozdělování a vyptávání času:

- Zaručený přenos kritických dat ve velmi krátkých a přesných isochronních cyklech s konfigurovatelným časováním.
- Velmi přesná synchronizace všech síťových uzlů v řádu méně než mikrosekund.
- Přenosy méně kritických dat ve vyhrazeném asynchronním kanálu.

Protokol dosahuje 200µs cykly s odchylkami (šumem) menšími než 1 µs. Tento protokol je možné využívat s libovolnými standardními ethernetovými součástmi topologie. Všechny protokoly IP jako je TCP a vyšší protokoly, je možno nadále používat bez úprav. [13]

Protokol POWERLINK je v souladu s následujícími mezinárodními normami:

- IEEE 802.3 Fast Ethernet
- IEC 61748-2
- IEC 61158
- Protokoly IP (UDP, TCP atd.)
- Standardní profily zařízení: podle normy CANopen EN 50325-4
- Standardní ethernetové obvody, nejsou nutné specializované obvody ASIC
- IEEE 1588 pro synchronizaci domén v reálném čase (budoucí verze) [13]



Obrázek 2.12: Ethernet POWERLINK [13]

Kapitola 3

Popis komunikace ACS880 a B&R PLC pomocí FEPL-02 a Ethernet POWERLINK

Komunikace mezi automatem a pohonem probíhá přes již popsany adaptér FEPL-02. Komunikace probíhá na principu Master-Slave. Například (což platí i pro tento případ) PLC nebo industriální počítač fungují jako řídicí uzel (Master) a všechna ostatní zařízení fungují jako řízené uzly (Slave). Master definuje hodinový pulz pro synchronizaci všech zařízení. V průběhu jednoho hodinového cyklu, během kterého jsou adresovány všechny uzly, Master posílá žádosti o data do všech řízených uzlů.

Cyklus komunikace Ethernet POWERLINK se skládá ze tří fází. V rámci startovací fáze dojde k zaslání zprávy z Master stanice do všech Slavů, aby došlo k synchronizaci zařízení. V druhé fázi dochází k výměně dat. Ve třetí fázi dochází k přenosu velkých paketů, které nejsou deterministické. Například parametry. Komunikace se testuje buď pomocí displeje na pohonu nebo přímo ve vývojové aplikaci Automation Studiu. V tomto případě v Automation Studiu. V případě chyby nebo jakéhokoli problému adaptér zobrazí chybovou hlášku. Všechny tyto hlášky jsou popsány v dokumentaci adaptéru, včetně jejich stavového diagramu. Komunikace pomocí Ethernet POWERLINK začíná inicializačním procesem a poté se pokračuje podle těchto stavových diagramů. [12]

Pro komunikaci je nutno si vybrat z jednoho z komunikačních profilů (volba probíhá při nutném nastavování hodnot na pohonu, tato volba včetně nastavení ostatních parametrů pro řízení je popsána v praktické části). V této práci je použit komunikační profil - ABB Komunikační profil pro řízení pohonů, který je popsán včetně blokového schématu a popisu kontrolního a stavového slova níže. Podle schématu a popisu slov se poté motor programuje. V tomto stavovém diagramu je popsáno, které bity je třeba aktivovat, aby došlo k určité funkci motoru.

Komunikace pro řízení jako taková tedy probíhá pomocí kontrolního a stavového slova. Kontrolér posílá kontrolní slovo do pohonu přes adaptér FEPL-02. Pomocí nastavování různých bitů na různé hodnoty probíhá řízení. V tabulkách níže jsou popsány všechny používané bity obou slov. Tyto tabulky se odkazují na schéma komunikačního profilu pod nimi. Díky těmto tabulkám a schématu je tvořena logika programu, která je popsána v praktické části.

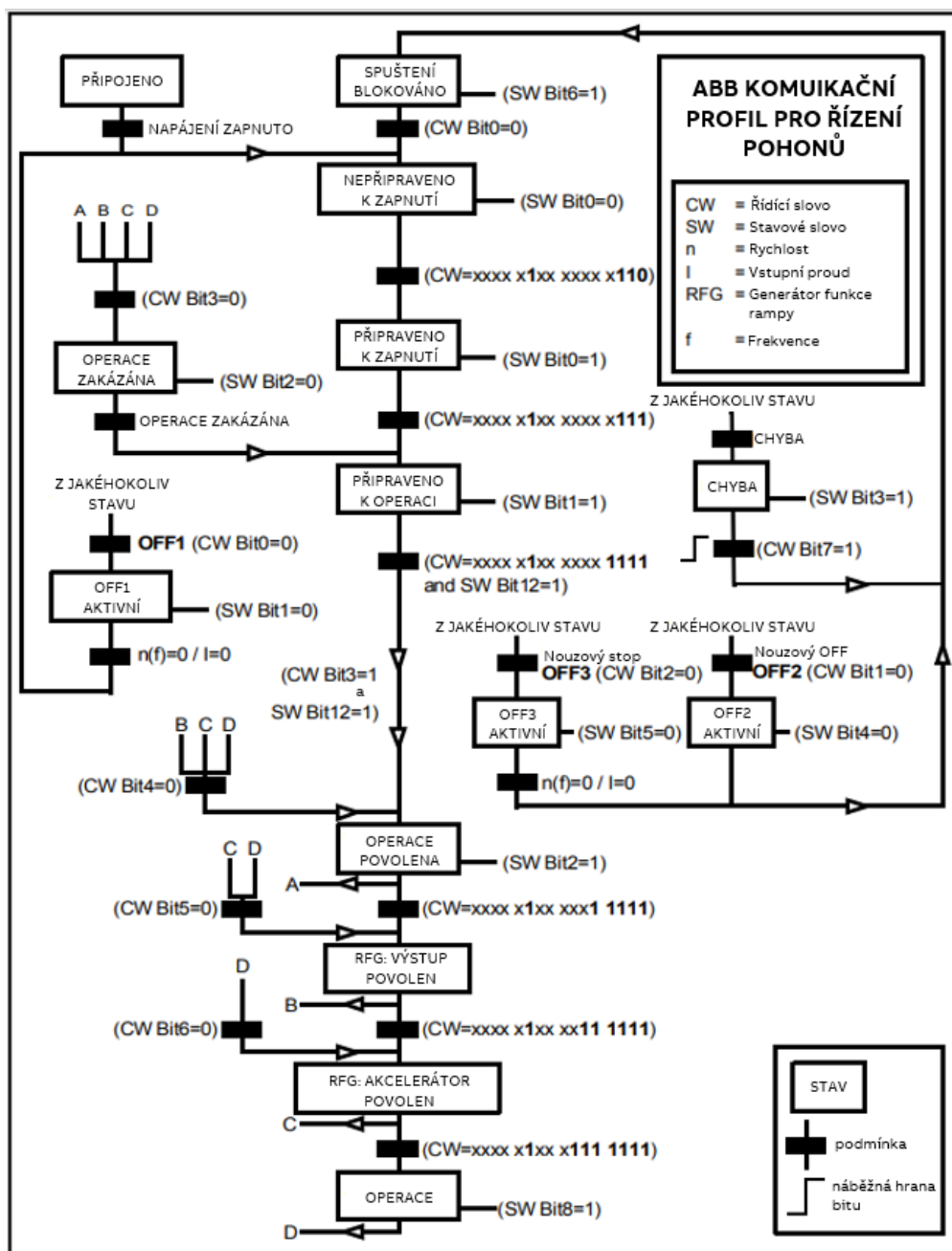
Bit	Název	Hodnota	Stav/Popis
0	OFF1_CONTROL	1	Pokračujte k PŘIPRAVENO K OPERACI .
		0	Zastavte aktivní zpomalení. Pokračujte k OFF1 AKTIVNÍ a k PŘIPRAVENO K ZAPNUTÍ dokud OFF2 a OFF3 jsou aktivní.
1	OFF2_CONTROL	1	Pokračující operace (OFF2 neaktivní).
		0	Nouzové vypnutí, okamžité zastavení. Pokračujte k OFF1 AKTIVNÍ , pokračujte k SPUŠTENÍ BLOKOVÁNO .
2	OFF3_CONTROL	1	Pokračující operace (OFF3 neaktivní).
		0	Nouzové vypnutí, zastavení v čase definované parametry pohonu. Pokračujte k OFF3 AKTIVNÍ , pokračujte k SPUŠTENÍ BLOKOVÁNO .
3	INHIBIT_OPERATION	1	Pokračujte k OPERACE POVOLENA .
		0	Operace blokována. Pokračujte k OPERACE BLOKOVÁNA .
4	RAMP_OUT_ZERO	1	Pokračujte k RFG: VÝSTUP POVOLEN .
		0	Změna výstupu funkčního generátoru na nulu. Pohon bude zastaven.
5	RAMP_HOLD	1	Povoluje rampu. Pokračujte k RFG: AKCELERÁTOR POVOLEN .
		0	Zastavení rampy.
6	RAMP_IN_ZERO	1	Normální operace. Pokračujte k OPERACE .
		0	Změna vstupu funkčního generátoru na nulu.
7	RESET	0 > 1	Resetování chyby pokud aktivní chyba existuje. Pokračujte k SPUŠTENÍ BLOKOVÁNO .
		0	Normální operace pokračuje.
8..9	Rezervováno.		
10	REMOTE_CMD	1	Dálkové řízení povoleno.
		0	Kontrolní slovo a jeho reference neprobíhají přes pohon.

Obrázek 3.1: Kontrolní slovo [12]

Bit	Název	Hodnota	Stav/Popis
0	RDY_ON	1	PŘIPRAVENO K SPUŠTĚNÍ
		0	NEPŘIPRAVENO K SPUŠTĚNÍ
1	RDY_RUN	1	PŘIPRAVENO K OPERACI
		0	OFF1 AKTIVNÍ
2	RDY_REF	1	OPERACE POVOLENA
		0	OPERACE BLOKOVÁNA
3	TRIPPED	1	CHYBA
		0	Bez chyby
4	OFF_2_STA	1	OFF2 neaktivní
		0	OFF2 AKTIVNÍ
5	OFF_3_STA	1	OFF3 neaktivní
		0	OFF3 AKTIVNÍ
6	SWC_ON_INHIB	1	SPUŠTENÍ BLOKOVÁNO
		0	-
7	ALARM	1	Varování/Alarm
		0	Žádné varování/alarm
8	AT_SETPOINT	1	OPERACE. Aktuální hodnota se rovná referenci.
		0	Aktuální hodnota se nerovná referenci.
9	REMOTE	1	Dálkové řízení
		0	Lokální řízení
10	ABOVE_LIMIT	1	Aktuální hodnota frekvence nebo rychlosti přesahuje limit (nastaven pomocí parametrů pohonu). Validní v obou směrech rotace.
		0	Aktuální hodnoty jsou v limitu.
15	FBA_ERROR	1	Komunikační chyba detekována adaptérem.
		0	Komunikace je v pořádku.

Obrázek 3.2: Stavové slovo [12]

Jak již bylo zmíněno, podle blokového schématu komunikačního profilu ABB, je pohon řízen. Na obr 3.3 je toho schéma k vidění. Lze z něj vypožorovat v jakém stavu se pohon nachází a to podle stavů jednotlivých bitů stavového slova - např. SW Bit2=0 což znamená, že dochází k blokování operace (**OPERACE ZAKÁZÁNA**). Popsání tohoto i všech dalších bitů je již popsáno v tabulkách výše. Pomocí kontrolního slova (také popsáno výše) dochází k přechodu mezi jednotlivými stavy pohonu. Například pro přechod ze stavu **SPUŠTĚNÍ BLOKOVÁNO** do **NEPŘIPRAVENO K ZAPNUTÍ** musí dojít k aktivaci CW Bit0 - což znamená, že musí být aktivován nultý bit kontrolního slova. Tímto způsobem je pohon řízen a je možné z něj číst jeho stavy.



Obrázek 3.3: Schéma komunikačního profilu pro řízení pohonu ACS880 [12]

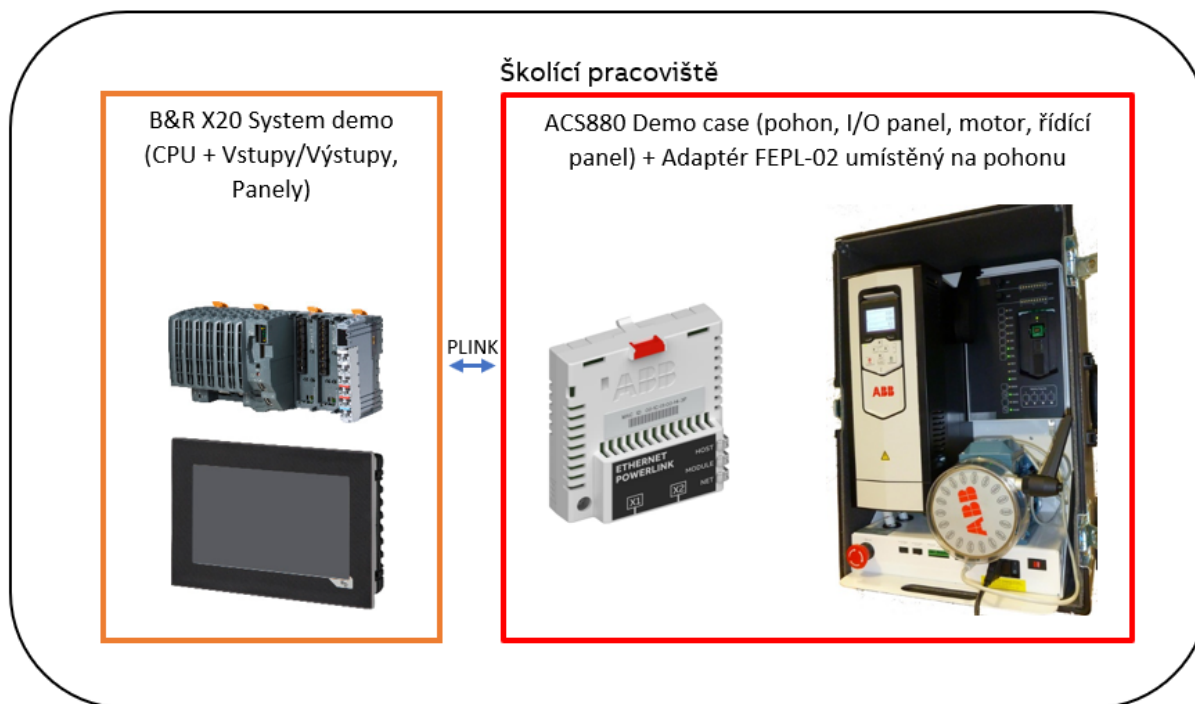
Kapitola 4

Návrh cílů pro testovací řídicí aplikaci za pomoci využití systému ACS880 a automatu B&R

Cílem této diplomové práce bylo vytvoření komunikace pomocí adaptéru FEPL-02 (komunikace v protokolu Ethernet POWERLINK) umístěném na pohonu ACS880 s B&R PLC a poté vytvoření řídicí - předváděcí aplikace pro řízení tohoto pohonu ACS880 pomocí tohoto automatu - X20CP1586, která bude využívat pro řízení a vizualizaci B&R panely T50 a T30. Komunikace probíhala za pomoci již zmíněného adaptéru, který umožnil komunikaci mezi PLC a pohonem a umožnil posílat kontrolní slovo a stavové slovo. Všechny tyto komponenty a jejich umístění a propojení v rámci tvořeného školícího pracoviště je možno vidět na obr. 4.1. Cílem bylo také popsání těchto konfigurací a jejich propojení - vše je popsáno v dalších kapitolách. Pomocí tohoto kontrolního a stavového slova je pak pohon řízen podle schématu využitého komunikačního profilu. Tato komunikace byla také testována pro správnou a požadovanou funkci. Vytvořená aplikace je používána k základnímu školení ohledně pohonů – kde je vysvětlen princip DTC, vliv zátěže na moment a celkově vlastnosti těchto pohonů. Dále je používána ke školení ohledně programování PLC od firmy B&R, včetně tvoření vizualizace.

K dispozici z B&R systému byly karty pro vstupy či výstupy, dva displeje – jeden byl použit čistě pro ovládací prvky a druhý pro grafy, tabulky a celkově aktuální referenční hodnoty. Dále samozřejmě procesor, jak již bylo zmíněno – X20CP1586. Řízení bylo tedy dosaženo pomocí PLC a aplikace bude řídit pohon ACS880. K dispozici bude také již zmíněná vizualizace na dvou displejích - jeden pro vizualizační prvky a druhý pro řídicí prvky.

Dalším cílem byla demonstrace a popis technologie DTC a celkově předvedení, jak se tyto pohony řídí, co je důležité měřit a sledovat na pohonech. Dále příklady větších pohonů na lodích případně nastínění technologie Azipod.



Obrázek 4.1: Návrh školícího pracoviště

Kapitola 5

Popis konfigurace systému a realizace programu pro navrženou aplikaci

V této kapitole je popsána konfigurace a topologie celého systému včetně realizace programu pro navrženou aplikaci za využití komponent v této konfiguraci.

5.1 Konfigurace

V této části je popsána konfigurace a seznam použitých komponent. V tabulce 5.1 je možno vidět názvy a označení komponent, jejich funkce včetně popisu co přesně v aplikaci mají za úkol. Rozdělení komponent je podle toho o jaké řízení se jedná - aplikace pracuje v režimu dálkového či lokálního řízení. Tyto režimy jsou popsány v dalších kapitolách.

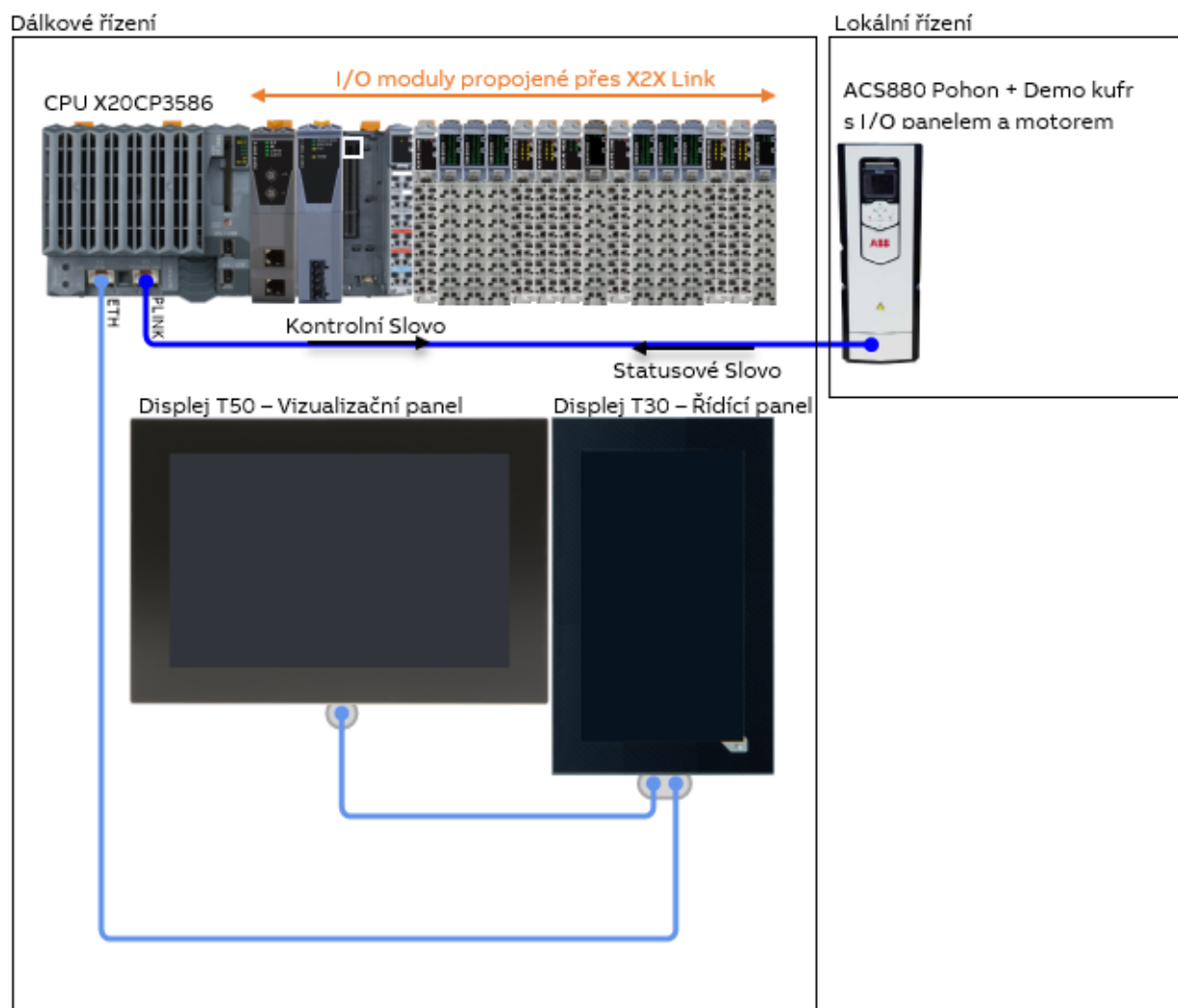
5.1.1 Seznam komponent a jejich popis včetně úlohy v aplikaci

Všechny komponenty v tabulce 5.1 jsou samostatné jednotky, které jsou spolu propojené. Poté je k dispozici Demo kufr, v kterém se nachází pohon ACS880, ovládaný motor a I/O panel s panelem na pohonu. Jak spolu komunikují všechny komponenty, bude popsáno v části práce zabývající se topologií.

Tabulka 5.1: Dostupné komponenty obsažené v konfiguraci aplikace

Komponent	Funkce v práci	Popis
CPU - X20CP3586	Řízení aplikace	Velice výkonný procesor od B&R, který řídí pohon a přijímá informace z vizualizačních panelů.
HMI Displej - 6PPT50.121E-10A	Vizualizační panel	Vizualizační panel, na kterém jsou zobrazeny aktuální hodnoty proměnných včetně průběhů pro dálkové řízení.
HMI Displej - 6PPT30.101N-20B	Řídicí panel	Vizualizační panel, na kterém jsou ovládací prvky pro dálkové řízení aplikace.
I/O Karty	Namapování vstupů/výstupů	Souhrn vstupních a výstupních karet - využito pouze na demonstraci jejich funkce a napojení proměnných.
Pohon - ACS880	Řízený pohon	Řízený pohon pomocí HMI displejů a CPU.
I/O Panel + Řídicí panel	Vizualizační panel	Slouží k ovládací pohonu v případě lokálního řízení, včetně nastavení všech parametrů pohonu.
Motor - M2VA63B4	Poháněný motor	Motor poháněný pohonem ACS880.

Na obrázku lze vidět jak jsou jednotlivé komponenty konfigurace propojeny a přes jaké rozhraní spolu komunikují. Dále je na obrázku vidět, který prvek je součástí lokálního řízení a který dálkového. Jak lze vidět, dálkové řízení je zařízeno pomocí PLC, kde PLC posílá do pohonu kontrolní slovo, díky kterému je pohon řízen. Pohon mimo přijímání kontrolního slova posílá do PLC stavové slovo, pro vyčtení informací v jakém stavu se pohon nachází. Spouštěcí sekvence a zápis do kontrolního slova je popsán v podkapitole zabývající se popisem řídicí aplikace, konkrétně zmíněného dálkového řízení. Přes jaké rozhraní probíhá komunikace je popsáno v kapitole o topologii. Lokální řízení probíhá za pomoci I/O panelu a displeji umístěném na pohonu. Stejně jako u dálkového řízení je vše popsáno v dalších kapitolách.



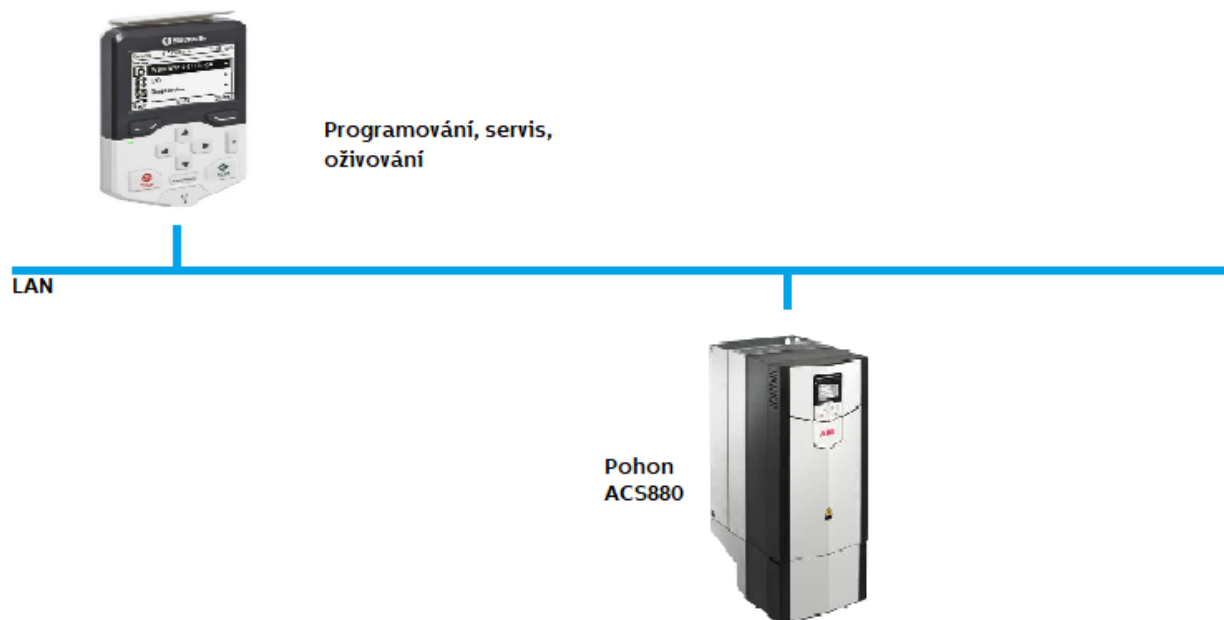
Obrázek 5.1: Konfigurace aplikace

5.2 Topologie

Zde je popsána topologie pro oba typy řízení. Topologie popisuje jak a přes jaké rozhraní spolu jednotlivé komponenty aplikace komunikují a který prvek topologie je nadřazen ostatním.

5.2.1 Lokální řízení

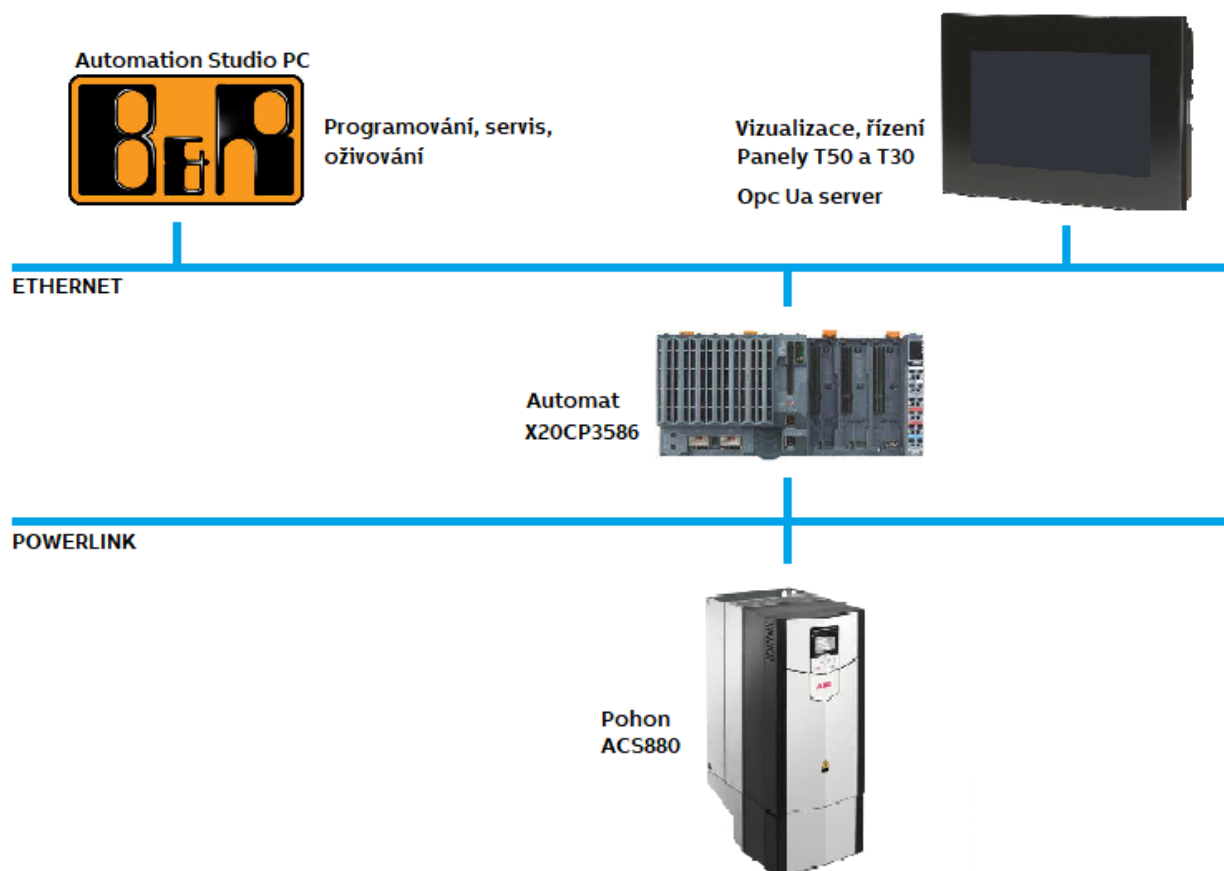
V případě lokálního řízení je topologie a komunikace jednoduchá. Pohon je součástí demo kufru jak již bylo řečeno a pohon komunikuje s I/O panelem a řídicím panelem (obr. 2.2) (umožňuje programování, servis a oživení pohonu), který je součástí demo kufru pomocí a propojen i s I/O panelem do lokální sítě - LAN, což lze vidět na obr. 5.2. Je zde také další možnost a to použití aplikace pro uvádění do provozu - Drive Composer. Tato aplikaci kompletně nahrazuje řídicí panel umístěný v I/O panelu pohonu. Tento displej je možno také umístit přímo na pohon - používá se v praxi na lodích kde je třeba konfigurovat přímo pohon, protože I/O panely jako takové, které jsou k dispozici zde jsou pouze pro předváděcí účely. Poté je topologie čistě Displej - Pohon. Displej je v takovém případě napojen přímo přes LAN konektor jak už bylo zmíněno - na pohon.



Obrázek 5.2: Topologie lokálního řízení

5.2.2 Dálkové řízení

V případě dálkové řízení je topologie a komunikace trochu složitější. Počítač, na kterém je aplikace programována komunikuje přes klasický Ethernet. To samé platí i pro propojení a komunikaci PLC s vizualizačními panely. Avšak co se týče komunikace mezi pohonem a PLC, zde je komunikace probíhá přes Ethernet POWERLINK. Komunikace mezi pohonem a PLC je přes tento protokol, protože komunikace přes klasické Ethernetové rozhraní není možná. Vysvětlení se nachází v kapitole zabývajícím se propojením těchto dvou komponentů - jak fyzické tak programové propojení. Jak jde na obr 5.3. vidět, PLC dostává pokyny z vizualizačních displejů, kam i zároveň posílá vyčtené aktuální referenční hodnoty z pohonu. Které hodnoty a do jakých vizualizačních prvků putují je popsáno v části práce zabývající se vizualizací.



Obrázek 5.3: Topologie dálkového řízení

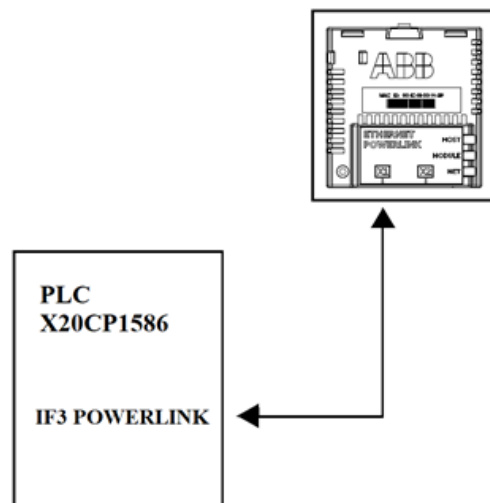
5.3 Propojení ACS880 s B&R automatem

Tato podkapitola se zabývá propojením pohonu s automatem pro jeho řízení. Propojení je popsáno jak fyzicky, tak programově.

5.3.1 Fyzické propojení

Původně se v pohonu nacházel adaptér FENA-01, který slouží pro komunikace přes rozhraní Ethernet. Ten byl vyměněn za potřebný adaptér FEPL-02, který slouží pro propojení s B&R systémem – komunikuje v protokolu Ethernet POWERLINK. Propojení je již znázorněno na obr. 5.4. Popis řízení pomocí tohoto adaptéru a samotného programového propojení je popsán v kapitole níže. Toto propojení bylo vybráno z důvodu, že propojení přes klasický Ethernet nebylo možné, z toho důvodu, že B&R pracuje v protokolu Ethernet POWERLINK. Když byla možnost propojení přes Ethernet testována, pohon nekomunikoval a to také z důvodu, že pro tento pohon jsou dostupné XDD soubory pouze pro Ethernet POWERLINK. Propojení v aplikaci a co je XDD soubor je popsáno v další části této kapitoly.

Propojení pohonu ACS880 a B&R PLC X20CP1586 je tedy provedeno přes Ethernet POWERLINK a pomocí adaptéru FEPL-02, který je umístěn v pohonu. Na obr. 5.4 je znázorněno propojení mezi již zmíněným PLC a adaptérem. Výstupem z PLC je POWERLINK, z kterého je kabel připojen na X1 slot na adaptéru. Slot X2 je na kartě v případě, že by bylo potřeba připojit více SLAVEů – například více pohonů do této POWERLINKové sítě. Komunikace je Master-Slave (Automat-Pohon). Po této výměně adaptéru a napojení kabelu z pohonu (adaptéru) do PLC - konkrétně na rozhraní č. 3 od PLC, bylo možno přejít k propojení v rámci aplikace.



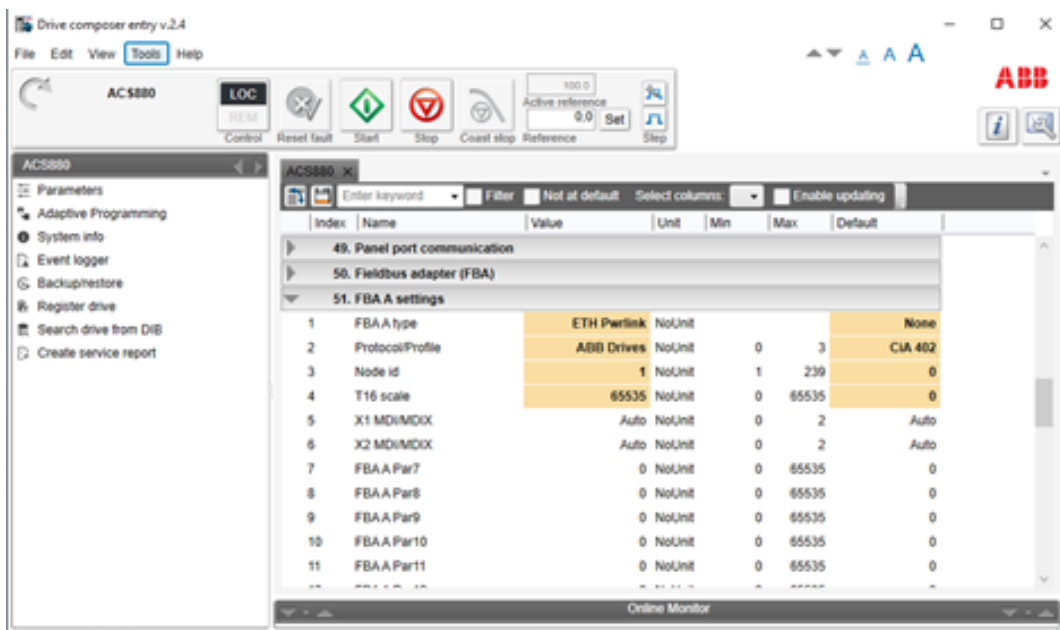
Obrázek 5.4: Schéma propojení automatu s pohonem [12]

5.3.2 Propojení v aplikaci

Po fyzickém připojení bylo zapotřebí provést nastavení parametrů v pohonu. Toto nastavení je provedeno proto, aby pohon dokázal komunikovat přes adaptér s automatem v určitém komunikačním protokolu - nastavení tedy umožňuje vytvořit komunikaci mezi komponenty. V pohonu jsou skupiny parametrů, které je potřeba nastavit - tyto parametry se mimo nastavení používají také k řízení pohonu, protože po propojení s automatem, budou všechny parametry co jsou k dispozici v pohonu (na řídicím panelu) k dispozici také v Automation Studio. Popis použití těchto hodnot je popsán opět v kapitole níže. Pro propojení však stačilo nastavit pouze malé množství těchto parametrů. Parametry se dají nastavit přímo na řídicím panelu umístěném na pohonu - obr. 5.5. nebo v aplikaci Drive Composer - obr. 5.6. Na tomto displeji je poté provedena i vizualizace včetně ovládání pro lokální řízení.



Obrázek 5.5: Nastavení parametrů na řídicím panelu, umístěném na pohonu



Obrázek 5.6: Nastavení parametrů v aplikaci Drive Composer

V následujících tabulkách je popsáno již zmíněné nastavení parametrů a co jaký parametr provádí. Parametry jsou řazeny do skupin a podskupin podle jejich vlastností.

Skupina parametrů 50:

Tabulka 5.2: Parametry pro řízení pohonu - Skupina 50 a její podskupiny[12]

	Jméno parametru	Hodnota	Popis
01	FBA A enable	1 = Enable	Povoluje komunikaci mezi pohonem a adaptérem FEPL-02.
02	FBA A comm loss func	1 = Fault	Povoluje monitoring chyb.
03	FBA A comm loss t out	3.0 s	Time out pro monitoring chyb.
04	FBA A ref1 type	4 = Speed	Nastavení hodnoty, která se sleduje. V tomto případě rychlost.

Skupina parametrů 51:

Tabulka 5.3: Parametry pro řízení pohonu - Skupina 51 a její podskupiny[12]

	Jméno parametru	Hodnota	Popis
01	FBA type	136 = EHT Powerlink	Nastavení protokolu, který je využit.
02	Profile	1 = ABB Drives profile	Profil, který je využit pro řízení pohonu.
03	Node ID	1	Číslo zařízení.
04	T16 Scale	65535	Měřítka pro převedení hodnoty na pohonu na RPM. Při takto nastavené hodnotě odpovídá 1 jedné otáčky za minutu. Platí ovšem pouze pro hodnoty využívající T16 scale.
26	Restore def conf	0 = No / 1 = Yes	Parametr, který se využívá pro případné resetování nastavení pohonu.
27	FBA A/B par refresh	0 = Done / 1 = Refresh	Parametr pro aktualizaci nastavených parametrů.

Skupina parametrů 20:

Tabulka 5.4: Parametry pro řízení pohonu - Skupina 20 a její podskupiny[12]

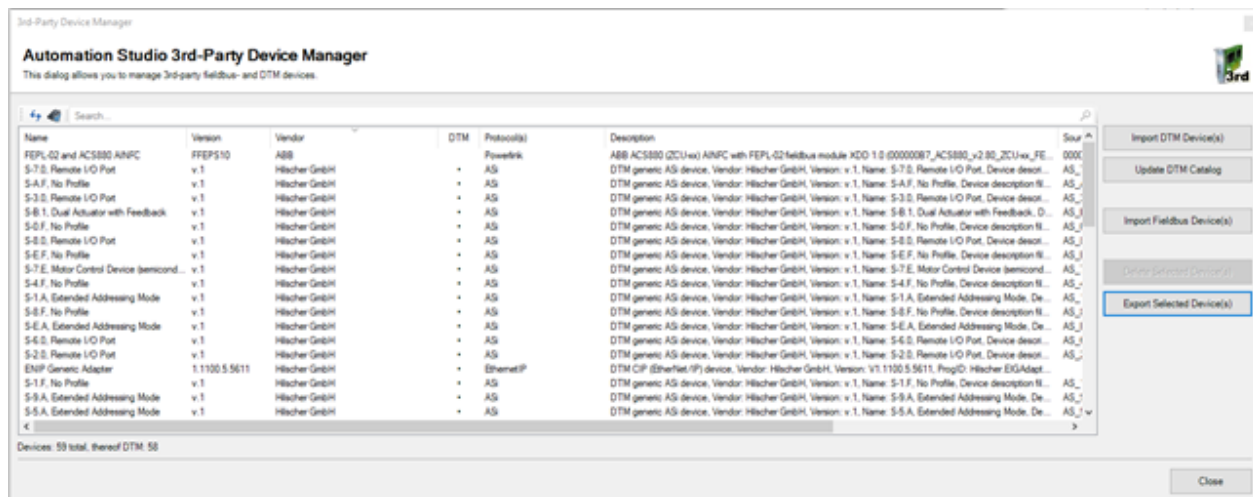
	Jméno parametru	Hodnota	Popis
01	Ext1 commands	12 = Fieldbus A	Volí fieldbus A interface jako zdroj start a stop příkazu.
02	Ext1 start trigger	1 = Level	Definuje start signál pro externí řízení.

Skupina parametrů 22:

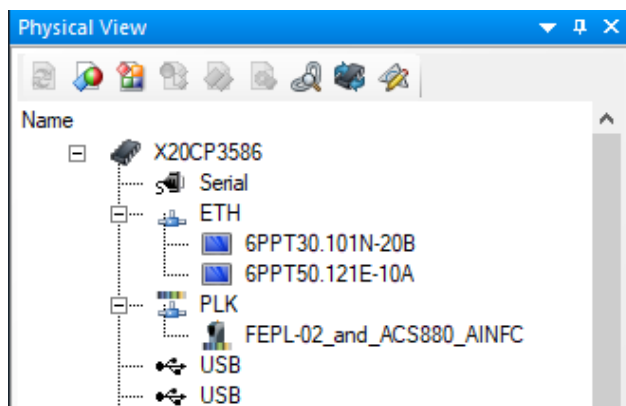
Tabulka 5.5: Parametry pro řízení pohonu - Skupina 22 a její podskupiny[12]

	Jméno parametru	Hodnota	Popis
11	Speed refl selection	4 = FB A refl	Volí fieldbus A interface jako zdroj start a stop příkazu.

Po nastavení všech parametrů, které jsou v tabulkách výše byla dalším krokem konfigurace master stanice – PLC. Potřebný byl XDD soubor pohonu, což je soubor, díky kterému je umožněna komunikace v B&R systému s jiným výrobcem – obdobné jako GSD soubor v SIEMENS. XDD soubor byl nahrán do Automation studia (obrázek 5.7) a pohon napojen na PLC – konkrétně na rozhraní č. 3 Ethernet POWERLINK (obrázek 5.8). Po propojení mohlo dojít ke kompilaci konfigurace a nahrání spojení do CPU.

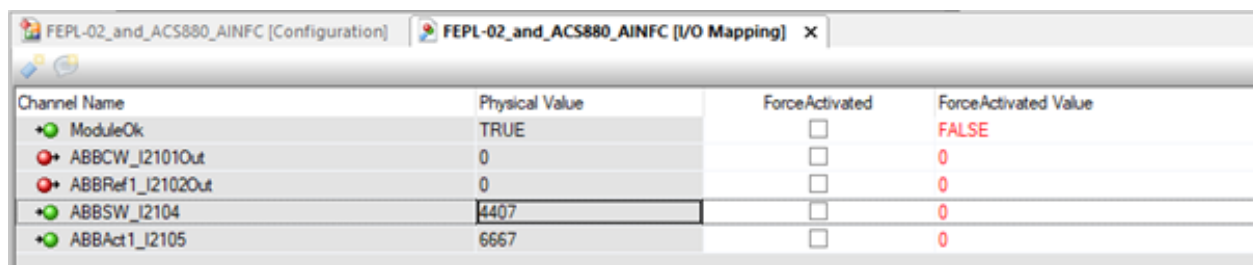


Obrázek 5.7: Nahrání XDD souboru do Automation Studia



Obrázek 5.8: Připojení pohonu na Ethernet POWERLINK

Po nahrání konfigurace do procesoru stačilo aplikaci spustit a v I/O monitoringu na pohonu ověřit, jestli je proměnná ModuleOK (modul v pořádku) v pohonu aktivní a propojení bylo dokončeno a ověřeno (obr. 5.9).



Channel Name	Physical Value	ForceActivated	ForceActivated Value
ModuleOk	TRUE	<input type="checkbox"/>	FALSE
ABBCW_I2101Out	0	<input type="checkbox"/>	0
ABBSW_I2104	4407	<input type="checkbox"/>	0
ABBA1_I2105	6667	<input type="checkbox"/>	0


Obrázek 5.9: Monitorování hodnot na pohonu

5.4 Tvorba a popis řídicí aplikace


V této části je popsána tvorba aplikace včetně jejího popisu. Jsou popsány použité proměnné, funkční bloky, v jaké cyklické třídě program běží, popis kontrolního a stavového slova včetně jeho přesného použití a principu, vysvětlení, která data putují odkud kam, alarmy, nastavení parametrů pohonu a hlavně je popsána myšlenka pro program zařizující dálkové řízení a také lokální řízení. Aplikace je rozdělená na tyto dva režimy, aby byl znázorněn rozdíl mezi lokálním a dálkovým řízením, že například v praxi se roboti dají ovládat přímo v jejich umístění, ale také dálkově a že mezi těmito režimy se dá přepínat a lze využívat najednou pouze jeden z těchto režimů. Přepínání těchto dvou režimů je možno na řídicím displeji umístěném v I/O panelu u pohonu.

5.4.1 Použité proměnné a funkční bloky

Proměnné, které využívají data z pohonu, jsou napojeny v konfiguraci na výstupy či vstupy právě z pohonu (obr. 5.10 a 5.11). Podrobněji popsáno využití kontrolního a stavového slova se nachází v další kapitole. V následujících tabulkách jsou popsány využití proměnné, včetně jejich datového typu a funkční bloky včetně jejich funkce v programu. V první tabulce jsou popsány čistě slova využitá pro řízení pohonu. V další tabulce jsou popsány proměnné a v poslední z tabulek funkční bloky.

 ABBCW_I2101Out	::Program:Control_Word	UINT
---	-------------------------------	-------------

Obrázek 5.10: Napojení vytvořené proměnné na hodnotu z pohonu - Vstup

 _0101_Motor_speed_used_I4001_S01	::Program:Speed_Used	DINT
---	-----------------------------	-------------

Obrázek 5.11: Napojení vytvořené proměnné na hodnotu z pohonu - Výstup

Tabulka 5.6: Proměnná pro kontrolní a stavové slovo s popisem jejich funkce v aplikaci

Název proměnné	Datový typ	Popis funkce
Control_Word	UINT	Kontrolní slovo, díky kterému je pohon řízen. Podrobnější popis je uveden v části práce zabývající se popisem řídicí aplikace.
Status_Word	UINT	Stavové slovo, díky kterému je automat schopen číst stav pohonu. Podrobnější popis je uveden stejně jako v případě kontrolního slova v další části práce.

Tabulka 5.7: Proměnné s popisem jejich funkce v aplikaci

Název proměnné	Datový typ	Popis funkce
Start_M	BOOL	Spuštění spouštěcí sekvence pohonu.
Stop_M	BOOL	Zastavení aplikace/pohonu.
Reset_Alarm	BOOL	Vy-resetování aktivních alarmů.
Negative_Speed	BOOL	Aktivace negativní rychlosti.
Reset_ASC	BOOL	Vy-resetování klopného obvodu zapínající sekvenci automatického řízení rychlosti.
Auto_Speed_Control	BOOL	Aktivace automatického řízení rychlosti.
Reset	BOOL	Reset všech proměnných a funkčních bloků do původního stavu.
LED_Module_OK	BOOL	Kontrolka ukazující jestli je propojení PLC s pohonem v pořádku.
LED_Ready_To_Operate	BOOL	Kontrolka, která znázorňuje, že je pohon připraven k operaci.
LED_Operation	BOOL	Kontrolka ukazující, že je pohon v chodu.
LED_Ready_To_Switch_ON	BOOL	Kontrolka ukazující, že je pohon připraven k sepnutí.
Speed_Used	DINT	Hodnota rychlosti čtená z pohonu.
Motor_Current	DINT	Hodnota proudu čtená z pohonu.
Motor_Temperature	DINT	Hodnota teploty čtená z pohonu.
Motor_Power	DINT	Hodnota výkonu čtená z pohonu.
Output_Voltage	DINT	Hodnota výstupního napětí čtená z pohonu.
Motor_Torque	DINT	Hodnota momentu čtená z pohonu.
Speed_IN	INT	Hodnota požadované rychlosti zadaná uživatelem.
Speed_Ref1	INT	Referenční hodnota rychlosti z pohonu.
Motor_Torque_Real	REAL	Hodnota momentu převedená do požadovaného formátu.
Speed_Used_Real	REAL	Hodnota rychlosti převedená do požadovaného formátu.
Motor_Power_Real	REAL	Hodnota výkonu převedená do požadovaného formátu.
Motor_Current_Real	REAL	Hodnota proudu převedená do požadovaného formátu.
Negative_Speed_ON	UDINT	Proměnná, do které je zapisován pokyn pro aktivaci negativní rychlosti.
Positive_Speed_ON	UDINT	Proměnná, do které je zapisován pokyn pro aktivaci pozitivní rychlosti.

Tabulka 5.8: Funkční bloky s popisem jejich funkce v aplikaci

Název funkčního bloku	Typ	Popis funkce
CTU_Start_Seq	CTU	Čítač směrem nahoru, využíván pro spuštění pohonu, zařizuje chod spouštěcí sekvence.
CTUD_Auto_Speed_Control	CTUD	Čítač směrem nahoru i dolů, využíván pro automatické zvyšování rychlosti v případě funkce automatického řízení rychlosti.
MpAlarmXCore_0	MpAlarmXCore	Funkční blok pro funkci alarmů v programu.
RS_ASC	RS	Klopný obvod, pro zapnutí či vypnutí automatického řízení rychlosti.
RS_Start_Stop	RS	Klopný obvod pro spouštění motoru.
RS_Reverse	RS	Klopný obvod pro povolení negativní rychlosti motoru.
TOF_ASC	TOF	Časovač se zpožděním při vypnutí, pro část programu řídící automatické řízení rychlosti.
TOF_CU	TOF	Časovač se zpožděním při vypnutí, pro generování pulzů do CTU_Start_Seq pro spuštění pohonu.
TON_CU	TON	Časovač se zpožděním při zapnutí, pro generování pulzů do CTU_Start_Seq pro spuštění pohonu.
TOF_ASC	TOF	Časovač se zpožděním při vypnutí, pro generování pulzů pro spouštění části programu řídící automatické řízení rychlosti.
TON_ASC	TON	Časovač se zpožděním při zapnutí, pro generování pulzů pro spouštění části programu řídící automatické řízení rychlosti.
TON_Delay_ASC	TON	Časovač se zpožděním při zapnutí, pro udržení maximální rychlosti motoru při automatickém řízení rychlosti.

5.5 Popis řídicí aplikace - Lokální řízení

Co se týče lokálního řízení, to probíhá za pomoci I/O panelu umístěném v demo kufru (obr. 5.12) a řídicím displeji umístěném v panelu. Na tomto displeji stejně jako v aplikaci Drive Composer, která byla využita, lze nastavit parametry potřebné pro správnou funkci. Pro toto řízení stačilo mít nastavené parametry, které byly nastaveny při propojení automatu s pohonem. Parametry týkající se regulátorů a dalších těchto technických parametrů byly přednastaveny výrobcem do ideálních hodnot pro co nejlepší funkci. Pohon s I/O panelem komunikuje v síti LAN, jak již bylo zmíněno v kapitole o topologii. Informace z panelu či displeje okamžitě putují do pohonu po zadání parametru, či spuštění motoru. Nastavení I/O panelu probíhá v displeji, stejně jako spouštění, nastavování rychlosti a jiných parametrů včetně vizualizace - ta bude popsána v kapitole o vizualizaci. Pohon se nachází v levé části kufru, následuje boční I/O panel a v něm umístěný řídicí panel (obr 2.2) a pod tímto panelem se nachází řízený motor. Na motoru je umístěná mechanická brzda, která slouží k zvyšování zátěže motoru - což bohužel nejde programově. To samé platí i pro aplikaci pro dálkové řízení.

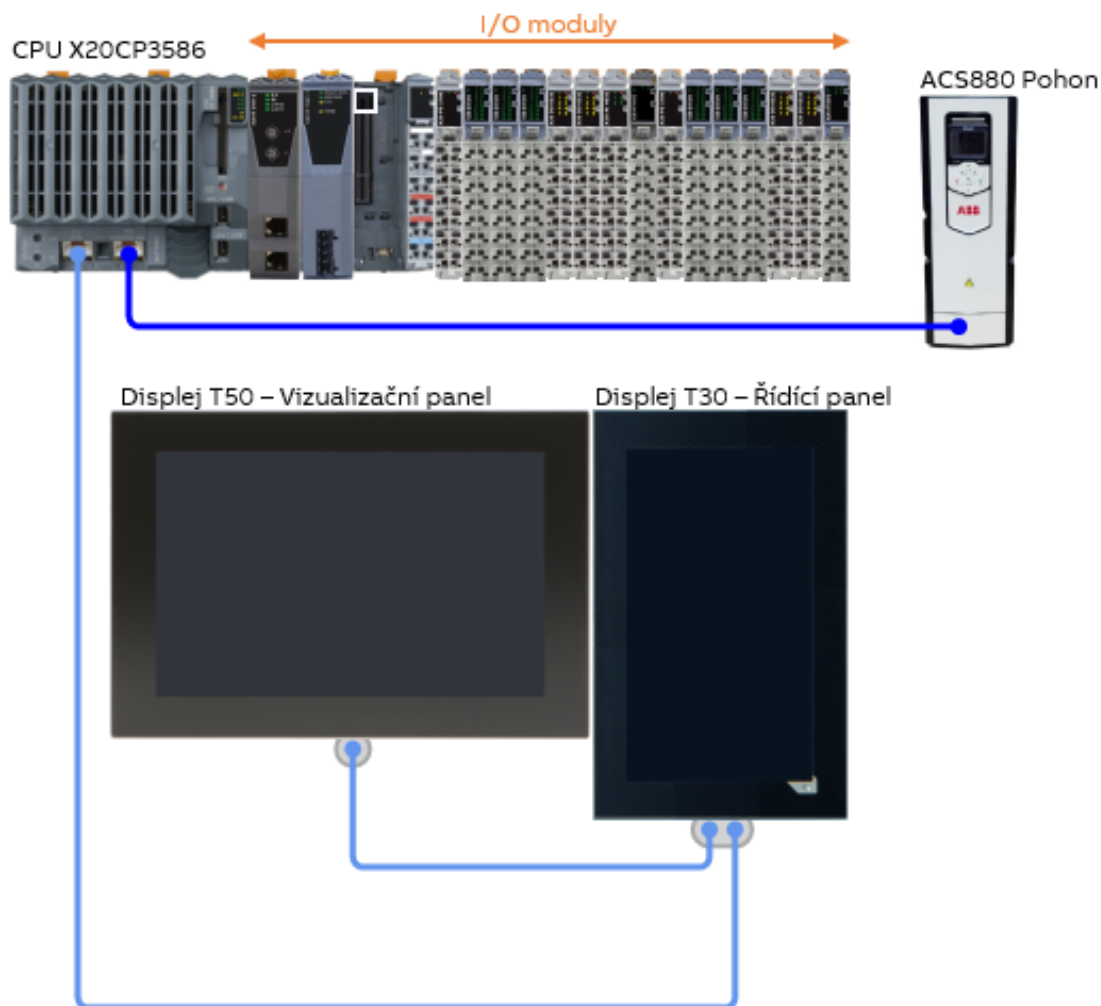


Obrázek 5.12: Konfigurace pro lokální řízení

5.6 Popis řídicí aplikace - Dálkové řízení

V případě dálkového řízení bylo třeba celou aplikaci vytvořit a naprogramovat. Hodnoty regulátorů však zůstávají přednastavené od výrobce pro co nejlepší funkci technologie Direct Torque Control. Cyklická třída (perioda s jakou se program provádí a dochází k načítání hodnot do proměnných a opačně) v které program probíhá je 10 ms. Je to nejrychlejší možná třída pro tento typ procesoru. Je zvolená z důvodu, že pohon dokáže provádět výpočty a funkci co 12.5 μs .

Na obrázku 5.13 je vidět, že aplikace funguje na principu posílání a přijímání kontrolního slova. PLC vyšle kontrolní slovo do pohonu, který ho zpracuje a poté se podle něho chová. Kontrolní slovo komunikuje podle zápisu binární hodnoty na jeho jednotlivé bity a to podle tabulky (obr. 3.1) uvedené v teoretické části. K spuštění pohonu, je potřeba se řídit stavovým diagramem, který je také uveden v teoretické části - konkrétně v části o komunikaci. Na něm je zjevně vidět (obr. 3.3), že motor nemůže být spuštěn pouze aktivováním vyhovujících bitů v kontrolním slovu na hodnoty k poslednímu stavu v diagramu (OPERACE), ale musí dojít k aktivaci bitů popořadě podle tohoto stavového diagramu. Proto je základem aplikace spouštěcí sekvence, kde v příslušném pořadí popsaném níže dochází k aktivaci bitů tak, aby došlo k onomu spuštění motoru.



Obrázek 5.13: Konfigurace pro dálkové řízení

Pod tímto odstavcem je popsán obsah jednotlivých částí sekvence. Celkem jsou 3. V první části dojde k povolení všech parametrů (výstup, akcelerátor a rampa povolena) kromě spuštění motoru - to je v první části sekvence blokováno. Druhá část sekvence povolí operaci a zapnutí. Poslední část sekvence je část, kdy dojde k spuštění motoru. Jednotlivé kroky sekvence se vykonávají s periodou 1 s ať nedojde k žádné kolizi. Spuštění sekvence či zastavení je zařízeno tlačítky Start či Stop. V případě jakékoliv nežádané kolize či nefunkčnosti, je k dispozici tlačítko Reset, pro nastavení všech proměnných do původních stavů.

Označení (index) jednotlivých bitů kontrolního slova:

$$1_{10}0_90_81_71_61_51_41_31_21_11_0$$

Bity jsou označeny v pořadí podle jejich indexu výše 10. bit = 10 atd.

1. část sekvence - Výstup Povoleno, Akcelerátor povolen, Spuštění blokováno:

$$4FF_{16} = 1279_{10} = 1001111111_2$$

- 10 - Dálkové řízení povoleno
- 9 - *Rezervováno*
- 8 - *Rezervováno*
- 7 - **Spuštění blokováno**
- 6 - Normální operace
- 5 - Rampa povolena
- 4 - Výstup povolen
- 3 - Operace není blokována
- 2 - Pokračující operace - OFF3 neaktivní
- 1 - Pokračující operace - OFF2 neaktivní
- 0 - Připraveno k operaci

2. část sekvence - Připraveno k zapnutí:

$$47E_{16} = 1150_{10} = 10001111110_2$$

- 10 - Dálkové řízení povoleno
- 9 - *Rezervováno*
- 8 - *Rezervováno*
- 7 - **Operace povolena**
- 6 - Normální operace
- 5 - Rampa povolena
- 4 - Výstup povolen
- 3 - Operace není blokována
- 2 - Pokračující operace - OFF3 neaktivní
- 1 - Pokračující operace - OFF2 neaktivní
- 0 - **Připraveno k zapnutí**

3. část sekvence - Operace:

$$47F_{16} = 1151_{10} = 10001111111_2$$

10 - Dálkové řízení povoleno

9 - *Rezervováno*

8 - *Rezervováno*

7 - **Operace povolena**

6 - Normální operace

5 - Rampa povolena

4 - Výstup povolen

3 - Operace není blokována

2 - Pokračující operace - OFF3 neaktivní

1 - Pokračující operace - OFF2 neaktivní

0 - **Připraveno k operaci**

Na obrázku 5.14 je ukázka části programu, která se zabývá spuštěním spouštěcí sekvence a tím pádem spuštěním motoru. Základem je spuštění či zastavení programu pomocí RS klopného obvodu. Dále následuje astabilní klopný obvod tvořen časovači TON a TOF, který generuje pulsy, které zařizují přičítání hodnot do čítače. Spouštěcí sekvence je jednoduše zařízena postupným posíláním hodnot do kontrolního slova podle hodnot na čítači.

```
//Základní start/stop logika
RS_Start_Stop.SET := Start_M;
RS_Start_Stop.RESET := Stop_pom OR Stop_M;
RS_Start_Stop();

IF Stop_M THEN
    Control_Word := 16#0;
END_IF;

//Astabilní klopný obvod pro generaci signálu do čítače
TON_CU.IN := RS_Start_Stop.Q1 AND NOT TOF_CU.Q;
TON_CU.PT := Time_1;
TON_CU();

TOF_CU.IN := TON_CU.Q;
TOF_CU.PT := Time_1;
TOF_CU();

//Čítač pro spouštěcí sekvenci motoru
CTU_Start_Seq.CU := TOF_CU.Q;
CTU_Start_Seq.PV := 4;
CTU_Start_Seq.RESET := Stop_pom OR CTU_Start_Seq.Q OR Stop_M;
CTU_Start_Seq();

IF CTU_Start_Seq.CV = 0 THEN
    Stop_pom := FALSE;
END_IF;

IF CTU_Start_Seq.CV = 1 THEN
    Control_Word := 16#4FF; //Výstup Povolen, Akcelerátor povolen, Spuštění blokováno
END_IF;

IF CTU_Start_Seq.CV = 2 THEN
    Control_Word := 16#47E; //Připraveno k zapnutí
END_IF;

IF CTU_Start_Seq.CV = 3 THEN
    Control_Word := 16#47F; //Operace
    Stop_pom := TRUE;
END_IF;
```

Obrázek 5.14: Část programu se základní spouštěcí sekvencí

Po spouštěcí frekvenci, bylo třeba zařídit, aby uživatel mohl zvolit hodnotu rychlosti v RPM (otáčky za minutu) a pohon se podle ní řídil. Pro tuto úlohu stačilo zapisovat do pohonu hodnotu rychlosti na jeho příslušnou proměnnou. Samozřejmě hodnotu bylo potřeba náležitě přepočítat, aby odpovídala skutečné zadané rychlosti. To samé platilo i pro volbu negativní rychlosti. Stačilo pouze zařídit aktivování proměnných, které tuto volbu v pohonu zařizují. Ovládací prvky pro tyto nastavení jsou dostupné na vizualizačních panelech - pro dálkové řízení a také na řídicím panelu pro lokální řízení. Všechny tyto prvky a nastavení jsou popsány v kapitole o vizualizaci níže.







Všechny proměnné a parametry, které se dají nastavovat na pohonu, mají nastavené limity podle řízeného pohonu. Ohledně vstupních hodnot do pohonu, například motor může dosahovat maximální hodnoty 1360 RPM, takže více než 1360 RPM nejde do motoru nastavit/poslat. Co se týče neřízených (výstupů z pohonu) hodnot (proud, napětí, teplota) ty jsou ošetřeny pomocí alarmů, které jsou popsány v podkapitole níže.

5.6.1 Automatické řízení rychlosti

Součástí aplikace je také sekvence, která zařizuje automatické navyšování rychlosti z nulové do maximální hodnoty s určitou periodou (2s), kdy se v maximální hodnotě ponechá po dobu 10s a poté se zase snižuje až na nulovou hodnotu rychlosti a motor se zastaví. Tato sekvence je v aplikaci dostupná z důvodu zobrazení průběhů všech sledovaných hodnot v celém spektru dovolených rychlostí. Jak již bylo řečeno, moment motoru programově měnit nejde a proto je zde možnost mechanické brzdy a pomocí ní moment zvyšovat, pro zobrazení sledovaných hodnot pod zátěží. Pro tuto část aplikace je použita stejná spouštěcí logika jako u programu bez tohoto režimu, doplněná o logiku navíc, avšak se stejnými typy funkčních bloků.

5.6.2 Alarmy

Jak již bylo naznačeno, alarmy jsou v aplikaci z důvodu, aby se žádná hodnota nedostala mimo její limity. Příklad nastavení alarmu je možno vidět na obr. 5.15. Je na něm vidět název alarmu, zpráva která se zobrazí ve vizualizaci v seznamu alarmů, proměnná, která se sleduje a její limit. Limit je nastaven na 1300, což je nižší hodnota než ta maximální a to z toho důvodu, aby uživatel byl včas varován. Toto nastavení platí pro všechny ostatní sledované a používané hodnoty.

	Alarm: SpeedLimit		
	Name	SpeedLimit	
	Message	Speed is: {&LimitText} {&Speed_Used_Real}	
	Code	222	
	Severity	1	
	Behavior	Level Monitoring	
	Monitoring		
	Monitored PV	::Program:Speed_Used_Real	Units
	Low Limit	Disabled	
	High Limit	Static	
	Limit	1300	Units
	Limit Text	High	
	HighHigh Limit	Disabled	

Obrázek 5.15: Příklad alarmu a jeho nastavení v aplikaci

Kapitola 6

Návrh a vytvoření vizualizace pro řízení pohonu

Tato kapitola se zabývá tvořením a popisem vizualizace pro navrženou aplikaci. Je v ní popsána konfigurace, komunikace a oba vizualizační displeje, včetně prvků, které využívají.

6.1 Konfigurace

Pro vizualizaci v rámci dálkového řízení jsou k dispozici dva displeje, zmíněné již v konfiguraci v rámci aplikace a topologie. Displeje jsou napojeny na PLC pomocí rozhraní Ethernet. Vizualizace posílá pokyny z ovládacích prvků do automatu a ten do vizualizace z pohonu zasílá referenční hodnoty, které se na displeji zobrazují. Oba tyto displeje jsou nastaveny na webový server (na jehož bázi pracují), který je složen z adresy PLC a názvu vizualizace. Poté je ještě nastavena adresa automatu a to stejně jako při nastavování adresy Ethernetu/automatu. Všechna tato nastavení probíhají jak v Automation Studio, tak přímo v panelech. V případě lokálního řízení je k dispozici I/O panel a řídicí displej - ty s pohonem komunikují v síti LAN. Oba tyto displeje jsou nastaveny na webový server, který je složen z adresy PLC a názvu vizualizace.

6.1.1 Komunikace - OpcUa Server

OpcUa server umožňuje komunikaci mezi vizualizací a automatem. Je to takřka jejich prostředník. V nastavení tohoto serveru se také povolují hodnoty, které mají být viditelné pro vizualizaci. Jsou to všechny hodnoty co jsou zapsány v tabulce výše. Konfigurace je před připravena v každém projektu Automation Studia, stačí ho jen povolit a propojení funguje.

Pracuje se stejnou adresou jako simulace, v které jde simulovat jak aplikace, tak vizualizační panely přímo ve webovém prohlížeči. Zobrazit tyto vizualizační panely jde zobrazit v prohlížeči také v případě normálního chodu aplikace v automatu. Stačí být pouze připojen přes Ethernet k počítači a v něm stačí mít k dispozici webový prohlížeč. Poté může dojít k zobrazení panelů ve webovém rozhraní stejně jako na panelu a to z toho důvodu, že panely pracují na principu zobrazení právě webového serveru. Adresa pro zobrazení je poté stejná jako adresa v panelech.

6.2 Sledované a řízené proměnné díky aplikaci a její vizualizaci

V tabulce níže je souhrn proměnných, které slouží ve vizualizaci jako vstupy do pohonu a také které slouží jako výstupy z pohonu jako sledované proměnné. Prostředníkem mezi vizualizací a pohonem je PLC. Důležité je připomenout, že vstupní hodnoty Start, Stop, Reset a Reset Alarm pouze spouštějí či resetují stavy pohonu přes kontrolní slovo. Tlačítka Automatické řízení rychlosti a Negativní otáčky zase pouze mění režimy řídicí aplikace. Nejedná se tedy přímo o vstupy do pohonu, ale vstupy do automatu, který je pak zpracuje a do pohonu pošle v rámci kontrolního Slova. Co se však týče výstupních hodnot z pohonu, ty jsou přímo hodnoty z pohonu, které tudíž můžeme nazvat výstupní hodnoty z pohonu.

Tabulka 6.1: Funkční bloky s popisem jejich funkce v aplikaci

"Vstupní proměnné" (název ve vizualizaci)	Výstupní proměnné (název ve vizualizaci)
Start_M (Start)	Motor_Current _{Real} (<i>Proud</i>)
Stop_M (Stop)	Output_Voltage (Napětí)
Reset (Reset)	Motor_Torque_Real (Moment)
Auto_Speed_Control (Automatické řízení rychlosti)	Speed_Used_Real (Rychlost)
Negative_Speed (Negativní otáčky)	Motor_Power_Real (Výkon)
Reset_Alarm (Alarm Reset)	Motor_Temperature (Teplota)
Speed_IN (Nastavení rychlosti)	LED_Module_OK (Propojení v pořádku)
	LED_Operation (Operace)
	LED_Ready_To_Operate (Připraveno k operaci)
	LED_Ready_To_Switch_On (Připraveno k sepnutí)

6.3 Vizualizace - Lokální řízení

Pro vizualizaci pro lokální řízení, je jak je možno vidět v konfiguraci k dispozici I/O panel s řídicím panelem v něm umístěný. Díky tomuto panelu je pohon řízen a monitorovány podstatné hodnoty nutné sledovat k správnému chodu motoru. Toto řízení je přítomno v těsné blízkosti pohonu. Popis řídicího panelu a jeho částí je k dispozici pod jeho obrázkem.

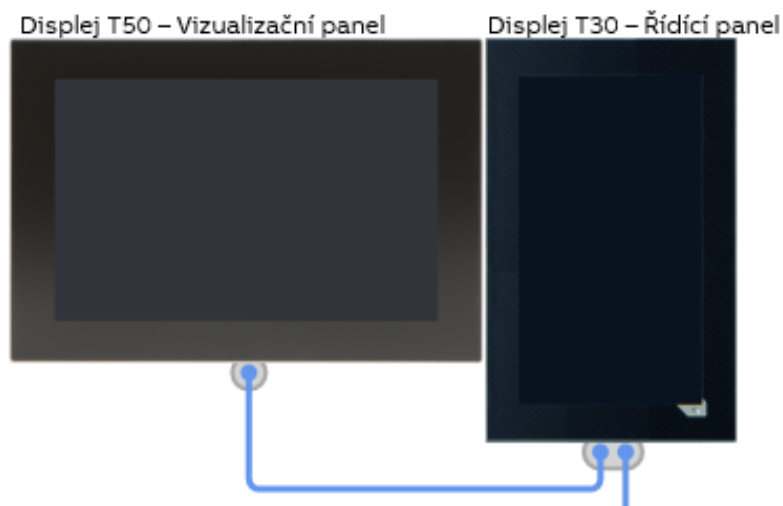


Obrázek 6.1: Vizualizace pro lokální řízení

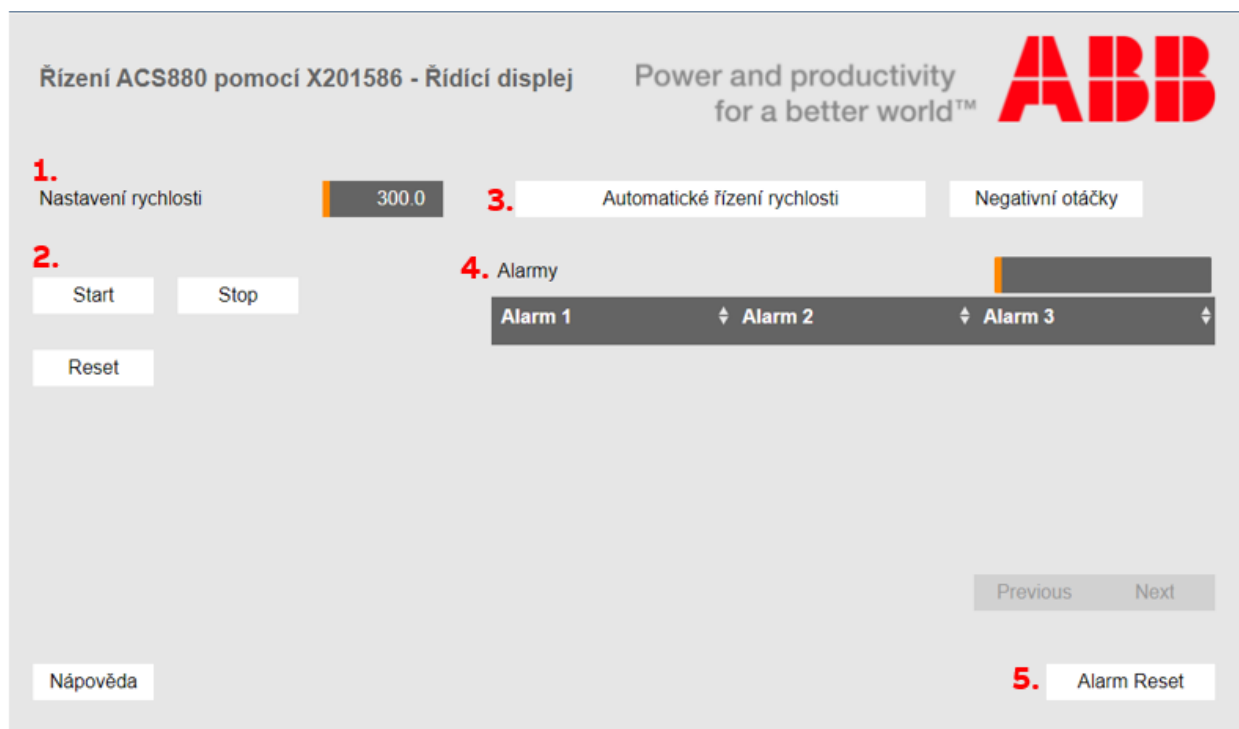
1. - Analogový výstup pro rychlost - -1360 - 1360 RPM
2. - Analogový výstup pro moment - -100% - 100%
3. - řídicí panel, na kterém lze vidět aktuální hodnoty motoru, změnit směr otáček a také vidět průběhy těchto sledovaných hodnot.
4. - Digitální vstup, který umožňuje chod pohonu.
5. - Tlačítková část řídicího panelu, kde dochází k spouštění či vypnutí motoru a také přepínání mezi lokálním a dálkovým řízením.

6.4 Vizualizace - Dálkové řízení

Pro dálkové řízení jsou k dispozici dva displeje. Jeden je určen pro řídicí prvky a druhý pro vizualizační. Díky těmto dvou displejům je tedy možno monitorování důležitých hodnot, stejně jako u lokálního řízení a pohon řídit. Popis obou těchto displejů je pod jejich obrázky níže.

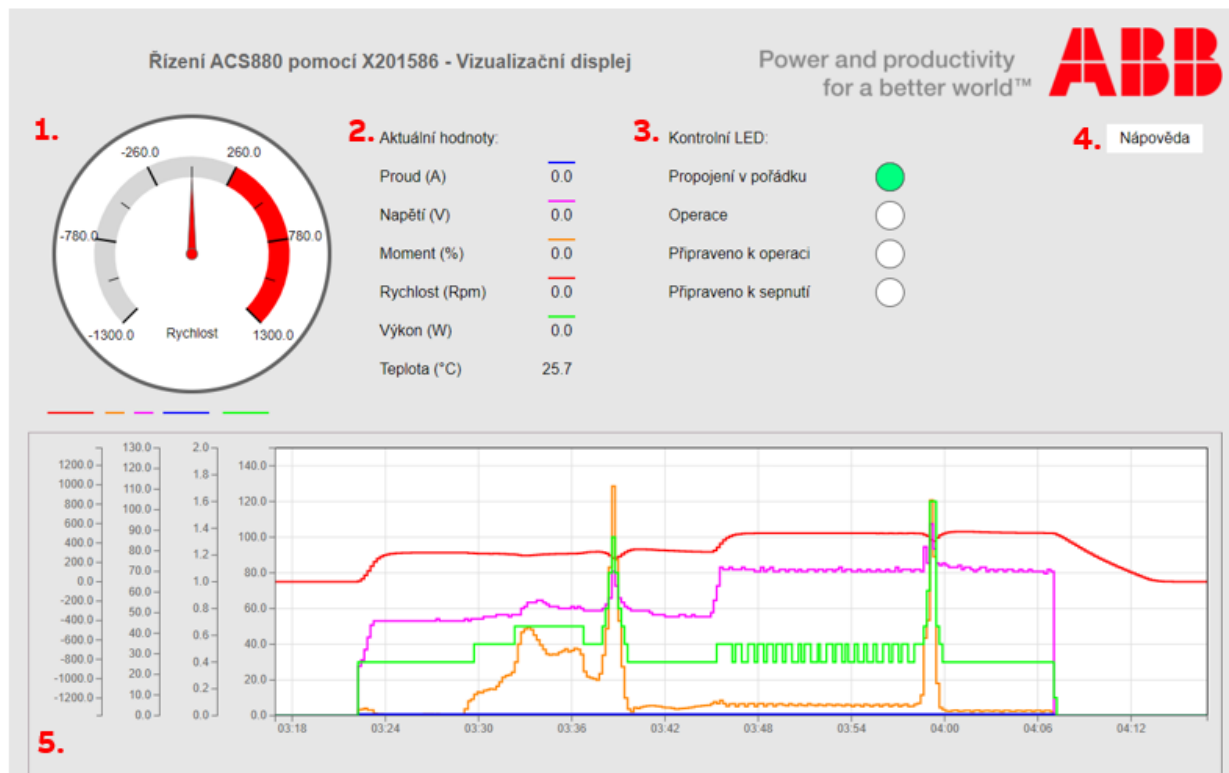


Obrázek 6.2: Konfigurace vizualizace pro dálkové řízení



Obrázek 6.3: Vizualizace pro dálkové řízení - řídicí displej

1. - Analogový vstup pro nastavení požadované rychlosti (možno měnit v průběhu chodu motoru).
2. - Část s ovládacími tlačítky - umožňuje spuštění, zastavení či vy-resetování pohonu/motoru.
3. - Část s tlačítky, které volí negativní otáčky či režim automatického řízení rychlosti.
4. - Seznam alarmů v případě, že k nějakému dojde.
5. - Tlačítko pro vymazání/odsouhlasení alarmů.



Obrázek 6.4: Vizualizace pro dálkové řízení - Vizualizační displej

1. - Budík pro analogový výstup rychlosti.
2. - Část s aktuálními hodnotami proměnných motoru.
3. - Část s kontrolními LED pro přehled v jakém stavu se pohon nachází.
4. - Náповěda jak pohon ovládat a co jaké tlačítko/ukazatele znamenají.
5. - Graf aktuálních hodnot na pohonu v čase.

Kapitola 7

Zhodnocení dosažených výsledků a možnosti uplatnění realizované aplikace

V této práci byly řešeny jak teoretické, tak praktické části. V teoretické části byly popsány využití komponenty, protokoly a systémy. Byl popsán ABB pohon ACS880, který byl řízen procesorem X20CP3586 z B&R X20 Systému. Z X20 systémy byly použity také dva displeje - jeden pro řízení a druhý pro vizualizaci. K propojení byl zapotřebí adaptér FEPL-02, díky kterému propojení těchto dvou prvků - procesor a pohon, mohlo dojít a kromě propojení umožnil také pohon řídit pomocí tohoto automatu. Dále byl kromě prvků již zmíněných, popsán také průmyslový protokol Ethernet POWERLINK, v jehož síti řízení probíhá.

Popsána byla také komunikace mezi těmito prvky, konkrétně komunikace mezi ACS880 a X20CP3586. Komunikace probíhala díky již zmíněného adaptéru FEPL-02. Tento adaptér umožnil propojení v síti Ethernet POWERLINK a řízení pomocí vybraného komunikačního profilu. Tento komunikační profil je řízen pomocí kontrolního a stavového slova jak je v práci uvedeno a popsáno.

Byly navrženy cíle, které byly popsání konfigurace systému, propojení systému s pohonem, vytvoření řídicí a školící aplikace pro ukázkou řízení pohonů, popis technologie DTC a shrnutí proměnných, které je třeba sledovat ve větších aplikacích s většími moduly těchto pohonů.

Kromě popsání komponentů v teoretické části, bylo dosaženo také vytvoření řídicí aplikace. Řídicí aplikace byla navržena do dvou režimů. Jeden klasické řízení rychlosti, kde uživatel reguluje rychlost sám a druhý, kde docházelo k automatickému řízení rychlosti od nejnížší do maximální rychlosti po určitých intervalech a o určitou hodnotu. Program byl navržen v strukturovaném textu a je ošetřen proti nežádoucím hodnotám vstupů či výstupů. V případě výstupu je to zařízeno pomocí alarmů, které jsou také přítomny. Dále byly použity režimy pro lokální a dálkové řízení. Dálkovým řízením je myšleno řízení pomocí PLC - a vizualizace. Tedy kdy X235CP86 řídí ACS880. V režimu pro lokální řízení je řízení zařízeno pomocí řídicího panelu a I/O panelu, který je připojen společně s pohonem v lokální síti. Tento panel je součástí demo kufru ACS880 společně s řízeným motorem.

K dispozici je vizualizace jak pro řízení, tak sledování důležitých proměnných. Řídicí displej tedy disponuje tlačítky a analogovými vstupy pro rychlost a vizualizační displej grafem, budíkem pro zobrazení rychlosti a analogové výstupy z pohonu/motoru. Kromě těchto ovládacích a vizualizačního prvků, je k dispozici také nápověda, kde je popsáno jak se má pohon řídit a co které tlačítko/prvek znamená.

Výsledkem je komplexní řídicí aplikace s řídicími panely, komunikující v protokolu Ethernet POWERLINK. Bude používána pro základní školení o X20 Systému, řízení pohonu ACS880, komunikace a konfigurace mezi těmito komponenty a nakonec je používána jako ukázka, jak se větší pohony, například využívající se na lodích řídí a jaké proměnné je třeba sledovat. Tyto proměnné jsou sledovány v vytvořené aplikaci. Popsána byla také technologie DTC - Direct Torque Control, kterou řízený pohon využívá. Byly splněny všechny cíle zadání a kromě používání aplikace ke školení bude u pohonu k dispozici také tato práce - jako jakýsi návod a základní informace o aplikaci a jejichž komponentech a taktéž návod na konfiguraci X20 System OMD demo, vytvořeného mnou a kolegy. V tomto návodu se opakují věci popsané v této práci, ale více do podrobnosti z teoretického hlediska včetně parametrů a popisu jednotlivých modulů. Toto demo je součástí sestavy, z které je používán procesor pro řízení této práce.

Literatura

1. *ABB industrial drives* [online] [cit. 2021-04-17]. Dostupné z: https://library.e.abb.com/public/adaccfb3b84a40b9bcaa3bc6746938c0/ACS880_Catalog_3AUA0000098111_Rev0_EN_17.2.2020.pdf.
2. *User's manual ACS880-01 democase* [online] [cit. 2021-04-17]. Dostupné z: https://library.e.abb.com/public/e1d9482227333703c1257aaa0044ef02/ACS880_01_democase_UM_A.pdf.
3. *Drive Composer* [online] [cit. 2021-04-17]. Dostupné z: <https://new.abb.com/drives/software-tools/drive-composer>.
4. *Motor - 3GVA062142-CSA* [online] [cit. 2021-04-17]. Dostupné z: <https://new.abb.com/products/3GVA062142-CSA/motor-with-shaft-extension-1430mm>.
5. *DTC - A motor control techniques for all seasons* [online] [cit. 2021-04-17]. Dostupné z: https://library.e.abb.com/public/0e07ab6a2de30809c1257e2d0042db5e/ABB_WhitePaper_DTC_A4_20150414.pdf.
6. *DTC vs PWM* [online] [cit. 2021-04-17]. Dostupné z: https://www.researchgate.net/profile/Zhengming_Zhao/publication/267556880/figure/fig5/AS:295545814962178/1447475021372/Ttorque-response-of-DTC-drive-compared-with-open-loop-PWM.png.
7. *Azipod* [online] [cit. 2021-04-17]. Dostupné z: <https://en.wikipedia.org/wiki/Azipod>.
8. *Users manual - X20 system* [online] [cit. 2021-04-17]. Dostupné z: https://download.br-automation.com/BRP44400000000000000639896/X20%20System-ENG_V3.60.pdf?px-hash=915e4585cd80d0aaef030a56ee8023b2&px-time=1619174375.
9. *C - programovací jazyk* [online] [cit. 2021-04-17]. Dostupné z: [https://cs.wikipedia.org/wiki/C_\(programovac%C3%AD_jazyk\)](https://cs.wikipedia.org/wiki/C_(programovac%C3%AD_jazyk)).
10. *Diagnostics* [online] [cit. 2021-04-17]. Dostupné z: <https://www.br-automation.com/cs/produkty/software/additional-information/diagnostics/>.
11. *System Diagnostics Manager - article* [online] [cit. 2021-04-17]. Dostupné z: <https://ifp1.com/are-your-service-calls-or-maintenance-adding-up-to-be-a-significant-cost/d>.

12. *FEPL-02 Ethernet POWERLINK adapter module* [online]. 2018 [cit. 2021-04-17]. Dostupné z: https://library.e.abb.com/public/1ac12cfb26894ebf8258af3ef6625798/EN_FEPL_02_UM_B_A4.pdf.
13. *Ethernet POWERLINK* [online] [cit. 2021-04-17]. Dostupné z: <https://www.br-automation.com/cs/produkty/networks-and-fieldbus-modules/additional-information/powerlink-technologie/>.

Příloha A

Řídící aplikace (Automation Studio)

Příloha v IS EDISON